Liaison 4.0

User Manual

Schrödinger Press

Revision A, April 2006

Contents

Document Conventions

In addition to the use of italics for names of documents, the font conventions that are used in this document are summarized in the table below.

Table 3.1.

Font	Example	Use
Sans serif	Project Table	Names of GUI features, such as panels, menus, menu items, buttons, and labels
Monospace	`$SCHRODINGER/maestro`	File names, directory names, commands, environment variables, and screen output
Italic	*filename*	Text that the user must replace with a value
Sans serif uppercase	CTRL+H	Keyboard keys

In descriptions of command syntax, the following UNIX conventions are used: braces { } enclose a choice of required items, square brackets [ ] enclose optional items, and the bar symbol | separates items in a list from which one item must be chosen. Lines of command syntax that wrap should be interpreted as a single command.

In this document, to *type* text means to type the required text in the specified location, and to *enter* text means to type the required text, then press the ENTER key.

References to literature sources are given in square brackets, like this: [10].

Introduction

1.1 About Liaison

Liaison™ predicts ligand-receptor binding affinities using a linear interaction approximation (LIA) model that has been fitted to a set known binding free energies. For each ligand in the training set, Liaison runs molecular mechanics (MM) simulations of the ligand-receptor complex at both endpoints of the binding process, bound ligand and free ligand. The simulation data and empirical binding affinities are analyzed to generate the Liaison parameters: α, β, and γ. These parameters are subsequently used to predict binding energies for other ligands with the same receptor. Liaison simulations use a continuum solvation model to shorten sampling times and speed convergence.

Liaison is run primarily from the Maestro™ graphical user interface. An introduction to Maestro is provided in Chapter 2 and a tutorial in using Liaison from Maestro appears in Chapter 3. Liaison can also be run from the command line, as described in Chapter 7. Utilities and scripts are run from the command line. Liaison technical notes, background, and references are provided in Chapter 8.

Maestro is Schrödinger's powerful, unified, multi-platform graphical user interface (GUI). It is designed to simplify modeling tasks, such as molecule building and data analysis, and also to facilitate the set up and submission of jobs to Schrödinger's computational programs. The main Maestro features include a project-based data management facility, a scripting language for automating large or repetitive tasks, a wide range of useful display options, a comprehensive molecular builder, and surfacing and entry plotting facilities. For more detailed information about the Maestro interface than is provided in Chapter 2, see the Maestro online help or the *Maestro User Manual*.

Protein Preparation is strongly recommended for protein and protein-ligand complex PDB structures to be used in Liaison. In most cases, this can be performed in Maestro, using the Protein Preparation panel in the Glide submenu. Command-line utilities complete the protein preparation facility. Protein preparation is described in Chapter 4; requirements for ligand structures are listed in Chapter 5.

The **Impact** computational program runs the MM calculations for Liaison simulations, which can be carried out using molecular dynamics (MD), hybrid Monte Carlo (HMC), or energy minimization. Impact uses an OPLS-AA force field. Impact calculations can also be run independently of Liaison. For more information, see the *Impact User Manual* and the *Impact Command Reference Manual*.

The Strike statistical analysis package is used for the fitting and prediction analysis tasks of Liaison. For more information, see the *Strike User Manual*.

1.2 Liaison Binding Energy Models

1.2.1 LIA Model Equation

A Liaison simulation combines a molecular-mechanics calculation with experimental data to build a model scoring function used to correlate or to predict ligand-protein binding free energies. The assumption used is that the binding energy can be approximated by comparing the energy of the bound complex with the energy of the free ligand-receptor system. A method of this type is called a Linear Response Method (LRM), a Linear Interaction Approximation (LIA), or a Linear Interaction Energy (LIE) method.

A novel feature of Liaison is that the simulation takes place in implicit (continuum) rather than explicit solvent—hence the name Liaison, for Linear Interaction Approximation in Implicit SOlvatioN. The explicit-solvent version of the methodology was first suggested by Aqvist (Hansson, T.; Aqvist, *J. Protein Eng.* **1995,** *8,* 1137-1145), based on approximating the charging integral in the free-energy-perturbation formula with a mean-value approach, in which the integral is represented as half the sum of the values at the endpoints, namely the free and bound states of the ligand. The empirical relationship used by Liaison is shown below:

$$\Delta G = \alpha\,(\langle U^{b}_{vdw}\rangle - \langle U^{f}_{vdw}\rangle) + \beta\,(\langle U^{b}_{elec}\rangle - \langle U^{f}_{elec}\rangle) + \gamma\,(\langle U^{b}_{cav}\rangle - \langle U^{f}_{cav}\rangle)$$

Here < > represents the ensemble average, *b* represents the bound form of the ligand, *f* represents the free form of the ligand, and α, β, and γ are the coefficients. U_{vdw}, U_{elec}, and U_{cav} are the van der Waals, electrostatic, and cavity energy terms in the Surface Generalized Born (SGB) continuum solvent model. The cavity energy term, U_{cav}, is proportional to the exposed surface area of the ligand. Thus, the difference:

$$\langle U^{b}_{cav}\rangle - \langle U^{f}_{cav}\rangle$$

measures the surface area lost by contact with the receptor. The net electrostatic interaction-energy in continuum solvent is given by:

$$U_{elec} = U_{coul} + 2\,U_{rxnf}$$

where U_{coul} is the Coulomb interaction energy and U_{rxnf} is the SGB-solvent reaction-field energy. (The factor of 2 compensates for the division by 2 made in the definition of the reaction-field free energy.)

In most applications, the coefficients α, β, and γ are determined empirically by fitting to the experimentally determined free energies of binding for a training set of ligands. In such appli-

cations, Liaison's simulation task is used to calculate the values of U_{vdw}, U_{elec}, and U_{cav} for the bound (complexed) and unbound (free) states of the training-set ligands, and its analysis task is used to derive values for the α, β, and γ fitting coefficients. The fitted equation can then be used to predict the binding affinities of additional ligands.

1.2.2 LiaisonScore Model Equation

Liaison also calculates a scoring function similar to GlideScore over the course of the LRM simulation. This scoring function, previously called "GlideScore in Liaison," is called LiaisonScore in Liaison 4.0. The average LiaisonScore can then be used to predict binding energies using the alternate model:

$$\Delta G = a(<LiaisonScore>) + b$$

where a is the LiaisonScore coefficient and b is a constant.

1.3 Documentation

For information related to the installation and use of Liaison and Impact, see the following documentation:

- The *Installation Guide*, which includes installation instructions for all Schrödinger products and documentation.
- The *Impact Command Reference Manual,* which contains syntax and keywords for Impact command input files.
- The *Maestro User Manual*, which describes how to use the features of Maestro, including the Atom Selection dialog box. An appendix describes command-line utilities, some of which may be used with Liaison.
- The *Maestro Command Reference Manual*, which contains commands, options, and arguments for running Maestro from the command line, including the Atom Specification Language (ASL) and the Entry Specification Language (ESL).

1.4 Citing Liaison in Publications

The use of this product and its components should be acknowledged in publications as:

Liaison, version 4.0, Schrödinger, LLC, New York, NY, 2005; Strike, version 1.5, Schrödinger, LLC, New York, NY, 2005.

Introduction to Maestro

Maestro is the graphical user interface for all of Schrödinger's products: CombiGlide™, Epik™, Glide™, Impact™, Jaguar™, Liaison™, LigPrep™, MacroModel®, Phase™, Prime™, QikProp™, QSite™, and Strike™. It contains tools for building, displaying, and manipulating chemical structures; for organizing, loading, and storing these structures and associated data; and for setting up, monitoring, and visualizing the results of calculations on these structures. This chapter provides a brief introduction to Maestro and some of its capabilities. For more information on any of the topics in this chapter, see the *Maestro User Manual*.

2.1 General Interface Behavior

Most Maestro panels are amodal: more than one panel can be open at a time, and a panel need not be closed for an action to be carried out. Each Maestro panel has a Close button so you can hide the panel from view.

Maestro supports the mouse functions common to many graphical user interfaces. The left button is used for choosing menu items, clicking buttons, and selecting objects by clicking or dragging. This button is also used for resizing and moving panels. The right button displays a shortcut menu. Other common mouse functions are supported, such as using the mouse in combination with the SHIFT or CTRL keys to select a range of items and select or deselect a single item without affecting other items.

In addition, the mouse buttons are used for special functions described later in this chapter. These functions assume that you have a three-button mouse. If you have a two-button mouse, ensure that it is configured for three-button mouse simulation (the middle mouse button is simulated by pressing or holding down both buttons simultaneously).

2.2 Starting Maestro

Before starting Maestro, you must first set the `SCHRODINGER` environment variable to point to the installation directory. To set this variable, enter the following command at a shell prompt:

csh/tcsh: `setenv SCHRODINGER` *installation-directory*

bash/ksh: `export SCHRODINGER=`*installation-directory*

You might also need to set the DISPLAY environment variable, if it is not set automatically when you log in. To determine if you need to set this variable, enter the command:

```
echo $DISPLAY
```

If the response is a blank line, set the variable by entering the following command:

csh/tcsh: `setenv DISPLAY` *display-machine-name*`:0.0`

bash/ksh: `export DISPLAY=`*display-machine-name*`:0.0`

After you set the SCHRODINGER and DISPLAY environment variables, you can start Maestro using the command:

```
$SCHRODINGER/maestro options
```

If you add the $SCHRODINGER directory to your path, you only need to enter the command maestro. Options for this command are given in Section 2.1 of the *Maestro User Manual*.

The directory from which you started Maestro is Maestro's current working directory, and all data files are written to and read from this directory unless otherwise specified (see Section 2.8 on page 27). You can change directories by entering the following command in the command input area (see page 8) of the main window:

```
cd directory-name
```

where *directory-name* is either a full path or a relative path.

2.3 The Maestro Main Window

The Maestro main window is shown in Figure 2.1 on page 7. The main window components are listed below.

The following components are always visible:

- **Title bar**—displays the Maestro version, the project name (if there is one) and the current working directory.
- **Auto-Help**—automatically displays context-sensitive help.
- **Menu bar**—provides access to panels.
- **Workspace**—displays molecular structures and other 3D graphical objects.

The following components can be displayed or hidden by choosing the component from the Display menu. Your choice of which main window components are displayed is persistent between Maestro sessions.

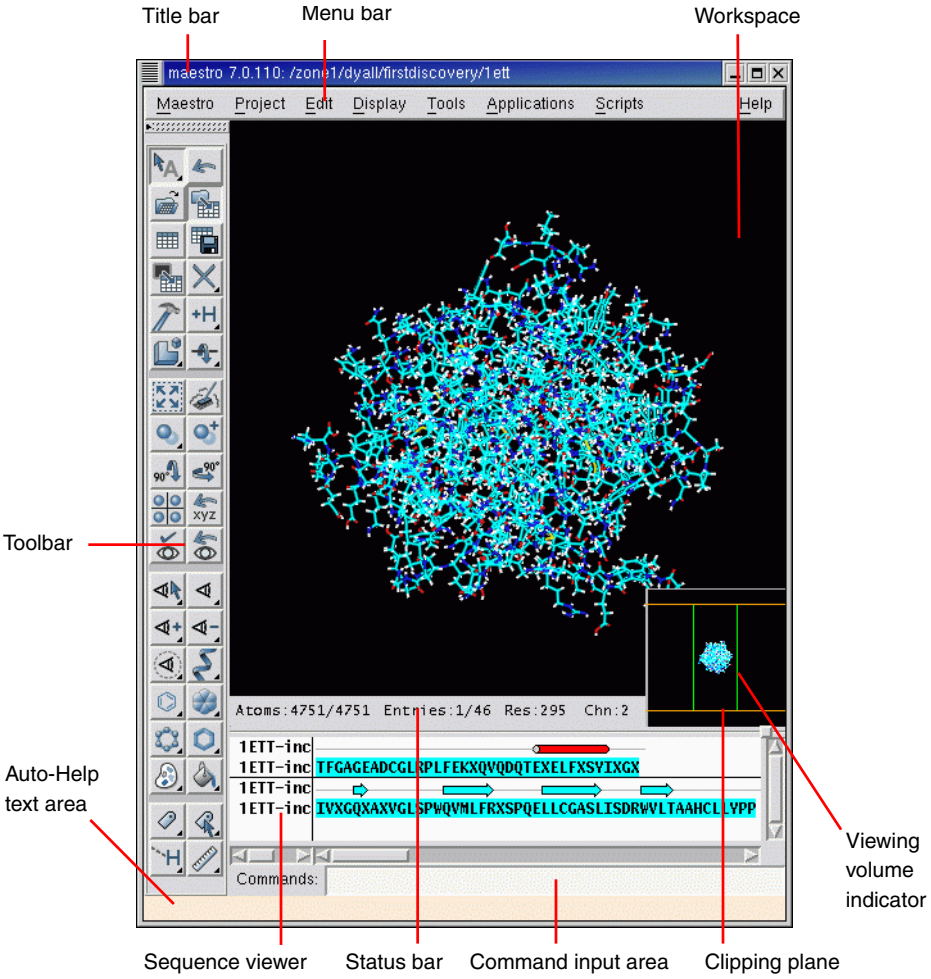


Figure 2.1. The Maestro main window.

- **Toolbar**—contains buttons for many common tasks and provides tools for displaying and manipulating structures, as well as organizing the Workspace.
- **Status bar**—displays information about a particular atom, or about structures in the Workspace, depending on where the pointer pauses (see Section 2.5 of the *Maestro User Manual* for details):
  - **Atom**—displays the chain, residue number, element, PDB atom name, formal charge, and title or entry name (this last field is set by choosing Preferences from the Maestro menu and selecting the Feedback folder).

  - **Workspace**—displays the number of atoms, entries, residues, chains, and molecules in the Workspace.

- **Clipping planes window**—displays a small, top view of the Workspace and shows the clipping planes and viewing volume indicators.
- **Sequence viewer**—shows the sequences for proteins displayed in the Workspace. See Section 2.6 of the *Maestro User Manual* for details.
- **Command input area**—provides a place to enter Maestro commands.

When a distinction between components in the main window and those in other panels is needed, the term *main* is applied to the main window components (e.g., main toolbar).

You can expand the Workspace to occupy the full screen, by pressing CTRL+=. All other components and panels are hidden. To return to the previous display, press CTRL+= again.

2.3.1 The Menu Bar

The menus on the main menu bar provide access to panels, allow you to execute commands, and control the appearance of the Workspace. The main menus are as follows:

- Maestro—save or print images in the Workspace, execute system commands, save or load a panel layout, set preferences, set up Maestro command aliases, and quit Maestro.
- Project—open and close projects, import and export structures, make a snapshot, and annotate a project. These actions can also be performed from the Project Table panel. For more information, see Section 2.4 on page 13.
- Edit—undo actions, build and modify structures, define command scripts and macros, and find atoms in the Workspace.
- Display—control the display of the contents of the Workspace, arrange panels, and display or hide main window components.
- Tools—group atoms; measure, align, and superimpose structures; and view and visualize data.
- Applications—set up, submit, and monitor jobs for Schrödinger's computational programs. Some products have a submenu from which you can choose the task to be performed.
- Scripts—manage and install Python scripts that come with the distribution and scripts that you create yourself. (See Chapter 13 of the *Maestro User Manual* for details.)
- Help—open the Help panel, the PDF documentation index, or information panels; run a demonstration; and display or hide Balloon Help (tooltips).

2.3.2 The Toolbar

The main toolbar contains three kinds of buttons for performing common tasks:

Action—Perform a simple task, like clearing the Workspace.

Display—Open or close a panel or open a dialog box, such as the Project Table panel.

Menu—Display a *button menu*. These buttons have a triangle in the lower right corner.

There are four types of items on button menus, and all four types can be on the same menu (see Figure 2.2):

- **Action**—Perform an action immediately.
- **Display**—Open a panel or dialog box.
- **Object types for selection**—Choose Atoms, Bonds, Residues, Chains, Molecules, or Entries, then click on an atom in the Workspace to perform the action on all the atoms in that structural unit.

  The object type is marked on the menu with a red diamond and the button is indented to indicate the action to be performed.
- **Other setting**—Set a state, choose an attribute, or choose a parameter and click on atoms in the Workspace to display or change that parameter.

The toolbar buttons are described below. Some descriptions refer to features not described in this chapter. See the *Maestro User Manual* for a fuller description of these features.

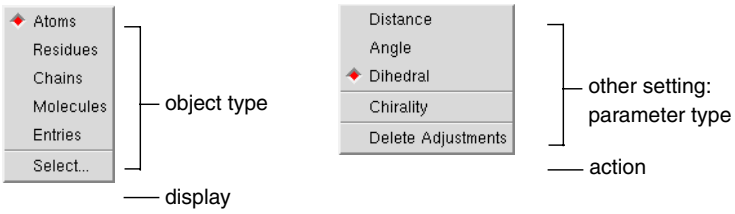


Figure 2.2. The Workspace selection ***button menu and the*** Adjust distances, angles or dihedrals ***button menu.***

Workspace selection
– Choose an object type for selecting
– Open the Atom Selection dialog box

Undo/Redo
Undo or redo the last action. Performs the same function as the Undo item on the Edit menu, and changes to an arrow pointing in the opposite direction when an Undo has been performed, indicating that its next action is Redo.

Open a project
Open the Open Project dialog box.

Import structures
Open the Import panel.

Open/Close Project Table
Open the Project Table panel or close it if it is open.

Save as
Open the Save Project As dialog box, to save the project with a new name.

Create entry from Workspace
Open a dialog box in which you can create an entry in the current project using the contents of the Workspace.

Delete
– Choose an object type for deletion
– Delete hydrogens and waters
– Open the Atom Selection dialog box
– Delete other items associated with the structures in the Workspace
– Click to select atoms to delete
– Double-click to delete all atoms

Open/Close Build panel
Open the Build panel or close it if it is open.

Add hydrogens
– Choose an object type for applying a hydrogen treatment
– Open the Atom Selection dialog box
– Click to select atoms to treat
– Double-click to apply to all atoms

Local transformation
– Choose an object type for transforming
– Click to select atoms to transform
– Open the Advanced Transformations panel

Adjust distances, angles or dihedrals
– Choose a parameter for adjusting
– Delete adjustments

Fit to screen
Scale the displayed structure to fit into the Workspace and reset the center of rotation.

Clear Workspace
Clear all atoms from the Workspace.

Set fog display state
Choose a fog state. Automatic means fog is on when there are more than 40 atoms in the Workspace, otherwise it is off.

Enhance depth cues
Optimize fogging and other depth cues based on what is in the Workspace.

Rotate around X axis by 90 degrees
Rotate the Workspace contents around the X axis by 90 degrees.

Rotate around Y axis by 90 degrees
Rotate the Workspace contents around the Y axis by 90 degrees.

Tile entries
Arrange entries in a rectangular grid in the Workspace.

Reset Workspace
Reset the rotation, translation, and zoom of the Workspace to the default state.

Save view
Save the current view of the Workspace: orientation, location, and zoom.

Restore view
Restore the last saved view of the Workspace: orientation, location, and zoom.

Display only selected atoms
– Choose an object type for displaying
– Click to select atoms to display
– Double-click to display all atoms

Display only
– Choose a predefined atom category
– Open the Atom Selection dialog box

Also display
– Choose a predefined atom category
– Open the Atom Selection dialog box

Undisplay
– Choose a predefined atom category
– Open the Atom Selection dialog box

Display residues within N angstroms of currently displayed atoms
– Choose a radius
– Open a dialog box to set a value

Show, hide, or color ribbons
– Choose to show or hide ribbons
– Choose a color scheme for coloring ribbons

Draw bonds in wire
– Choose an object type for drawing bonds in wire representation
– Open the Atom Selection dialog box
– Click to select atoms for representation
– Double-click to apply to all atoms

Draw atoms in CPK
– Choose an object type for drawing bonds in CPK representation
– Open the Atom Selection dialog box
– Click to select atoms for representation
– Double-click to apply to all atoms

Draw atoms in Ball & Stick
– Choose an object type for drawing bonds in Ball & Stick representation
– Open the Atom Selection dialog box
– Click to select atoms for representation
– Double-click to apply to all atoms

Draw bonds in tube
– Choose an object type for drawing bonds in tube representation
– Open the Atom Selection dialog box
– Click to select atoms for representation
– Double-click to apply to all atoms

Color all atoms by scheme
Choose a predefined color scheme.

Color residue by constant color
– Choose a color for applying to residues
– Click to select residues to color
– Double-click to color all atoms

Label atoms
– Choose a predefined label type
– Delete labels

Label picked atoms
– Choose an object type for labeling atoms
– Open the Atom Selection dialog box
– Open the Atom Labels panel at the Composition folder
– Delete labels
– Click to select atoms to label
– Double-click to label all atoms

Display H-bonds
– Choose bond type:
intra—displays H-bonds within the selected molecule
inter—displays H-bonds between the selected molecule and all other atoms.
– Delete H-bonds
– Click to select molecule

Measure distances, angles or dihedrals
– Choose a parameter for displaying measurements
– Delete measurements
– Click to select atoms for measurement

2.3.3 Mouse Functions in the Workspace

The left mouse button is used for selecting objects. You can either click on a single atom or bond, or you can drag to select multiple objects. The right mouse button opens shortcut menus, which are described in Section 2.7 of the *Maestro User Manual*.

The middle and right mouse buttons can be used on their own and in combination with the SHIFT and CTRL keys to perform common operations, such as rotating, translating, centering, adjusting, and zooming.

Table 2.1. Mapping of Workspace operations to mouse actions.

Mouse Button	Keyboard	Motion	Action
Left		click, drag	Select
Left	SHIFT	click, drag	Toggle the selection
Middle		drag	Rotate about X and Y axes Adjust bond, angle, or dihedral
Middle	SHIFT	drag vertically	Rotate about X axis
Middle	SHIFT	drag horizontally	Rotate about Y axis
Middle	CTRL	drag horizontally	Rotate about Z axis
Middle	SHIFT + CTRL	drag horizontally	Zoom
Right		click	Spot-center on selection
Right		click and hold	Display shortcut menu
Right		drag	Translate in the X-Y plane
Right	SHIFT	drag vertically	Translate along the X axis
Right	SHIFT	drag horizontally	Translate along the Y axis
Right	CTRL	drag horizontally	Translate along the Z axis
Middle & Right		drag horizontally	Zoom

2.3.4 Shortcut Key Combinations

Some frequently used operations have been assigned shortcut key combinations. The shortcuts available in the main window are described in Table 2.2.

Table 2.2. Shortcut keys in the Maestro main window.

Keys	Action	Equivalent Menu Choices
CTRL+B	Open Build panel	Edit > Build
CTRL+C	Create entry	Project > Create Entry From Workspace
CTRL+E	Open Command Script Editor panel	Edit > Command Script Editor
CTRL+F	Open Find Atoms panel	Edit > Find
CTRL+H	Open Help panel	Help > Help
CTRL+I	Open Import panel	Project > Import Structures
CTRL+M	Open Measurements panel	Tools > Measurements
CTRL+N	Create new project	Project > New
CTRL+O	Open project	Project > Open
CTRL+P	Print	Maestro > Print
CTRL+Q	Quit	Maestro > Quit
CTRL+S	Open Sets panel	Tools > Sets
CTRL+T	Open Project Table panel	Project > Show Table
CTRL+W	Close project	Project > Close
CTRL+Z	Undo/Redo last command	Edit > Undo/Redo
CTRL+=	Enter and exit full screen mode (Workspace occupies full screen)	None

2.4 Maestro Projects

All the work you do in Maestro is done within a *project*. A project consists of a set of *entries*, each of which contains one or more chemical structures and their associated data. In any Maestro session, there can be only one Maestro project open. If you do not specify a project when you start Maestro, a *scratch* project is created. You can work in a scratch project without saving it, but you must save it in order to use it in future sessions. When you save or close a project, all the view transformations (rotation, translation, and zoom) are saved with it. When you close a project, a new scratch project is automatically created.

Likewise, if there is no entry displayed in the Workspace, Maestro creates a *scratch* entry. Structures that you build in the Workspace constitute a scratch entry until you save the structures as project entries. The scratch entry is not saved with the project unless you explicitly add it to the project. However, you can use a scratch entry as input for some calculations.

To add a scratch entry to a project, do one of the following:

- Click the Create entry from Workspace button:

- Choose Create Entry from Workspace from the Project menu.
- Press CTRL+C.

In the dialog box, enter a name and a title for the entry. The entry name is used internally to identify the entry and can be modified by Maestro. The title can be set or changed by the user, but is not otherwise modified by Maestro.

Once an entry has been incorporated into the project, its structures and their data are represented by a row in the Project Table. Each row contains the row number, an icon indicating whether the entry is displayed in the Workspace (the In column), the entry title, a button to open the Surfaces panel if the entry has surfaces, the entry name, and any entry properties. The row number is not a property of the entry.

Entries can be collected into groups, and the members of the group can be displayed or hidden. Most additions of multiple entries to the Project Table are done as entry groups.

You can use entries as input for all of the computational programs—Glide, Impact, Jaguar, Liaison, LigPrep, MacroModel, Phase, Prime, QikProp, QSite, and Strike. You can select entries as input for the ePlayer, which displays the selected structures in sequence. You can also duplicate, combine, rename, and sort entries; create properties; import structures as entries; and export structures and properties from entries in various formats.

To open the Project Table panel, do one of the following:

- Click the Open/Close Project Table button on the toolbar

- Choose Show Table from the Project menu
- Press CTRL+T.

The Project Table panel contains a menu bar, a toolbar, and the table itself.

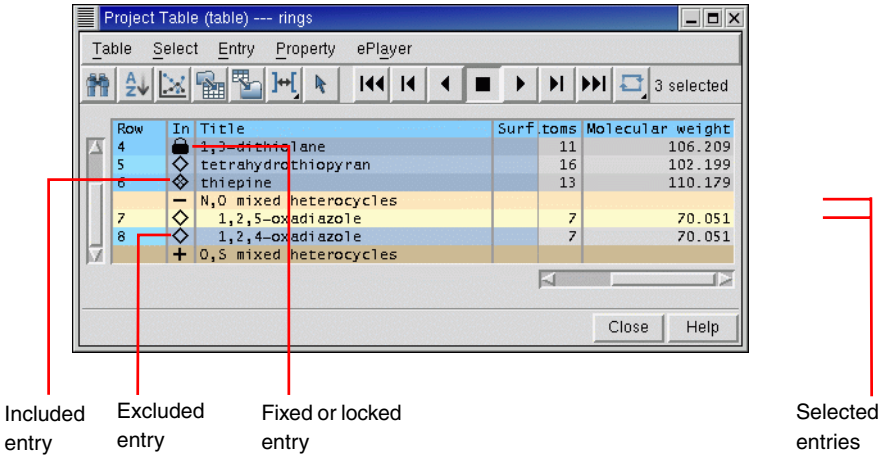


Figure 2.3. The** Project Table **panel.

2.4.1 The Project Table Toolbar

The Project Table toolbar contains two groups of buttons and a status display. The first set of buttons opens various panels that allow you to perform functions on the entries in the Project Table. The second set of buttons controls the ePlayer, which "plays through" the selected structures: each structure is displayed in the Workspace in sequence, at a given time interval. See Section 2.3.2 on page 9 for a description of the types of toolbar buttons. The buttons are described below.

Find
Open the Find panel for locating alphanumeric text in any column of the Project Table, except for the row number.

Sort
Open the Sort panel for sorting entries by up to three properties.

Plot
Open the Plot panel for plotting entry properties.

Import Structure
Open the Import panel for importing structures into the project.

Export Structure
Open the Export panel for exporting structures to a file.

Columns
Choose an option for adjusting the column widths.

Select only
Open the Entry Selection dialog box for selecting entries based on criteria for entry properties.

Go to start
Display the first selected structure.

Previous
Display the previous structure in the list of selected structures.

Play backward
Display the selected structures in sequence, moving toward the first.

Stop
Stop the ePlayer.

Play forward
Display the selected structures in sequence, moving toward the last.

Next
Display the next structure in the list of selected structures.

Go to end
Display the last selected structure.

Loop
Choose an option for repeating the display of the structures. Single Direction displays structures in a single direction, then repeats. Oscillate reverses direction each time the beginning or end of the list is reached.

The status display, to the right of the toolbar buttons, shows the number of selected entries. When you pause the cursor over the status display, the Balloon Help shows the total number of entries, the number shown in the table, the number selected, and the number included in the Workspace.

2.4.2 The Project Table Menus

- Table—find text, sort entries, plot properties, import and export structures, and configure the Project Table.
- Select—select all entries, none, invert your selection, or select classes of entries using the Entry Selection dialog box and the Filter panel.

- Entry—include or exclude entries from the Workspace, display or hide entries in the Project Table, and perform various operations on the selected entries.
- Property—display and manipulate entry properties in the Project Table.
- ePlayer—view entries in succession, stop, reverse, and set the ePlayer options.

2.4.3 Selecting Entries

Many operations in Maestro are performed on the entries selected in the Project Table. The Project Table functions much like any other table: select rows by clicking, shift-clicking, and control-clicking. However, because clicking in an editable cell of a selected row enters edit mode, you should click in the Row column to select entries. See Section 2.4.5 on page 18 for more information on mouse actions in the Project Table. There are shortcuts for selecting classes of entries on the Select menu.

In addition to selecting entries manually, you can select entries that meet a combination of conditions on their properties. Such combinations of conditions are called *filters*. Filters are Entry Selection Language (ESL) expressions and are evaluated at the time they are applied. For example, if you want to set up a Glide job that uses ligands with a low molecular weight (say, less than 300) and that has certain QikProp properties, you can set up a filter and use it to select entries for the job. If you save the filter, you can use it again on a different set of ligands that meet the same selection criteria.

To create a filter:

1. Do one of the following:
   - Choose Only, Add, or Deselect from the Select menu.
   - Click the Entry selection button on the toolbar.

   
2. In the Properties folder, select a property from the property list, then select a condition.
3. Combine this selection with the current filter by clicking Add, Subtract, or Intersect. These buttons perform the Boolean operations OR, AND NOT, and AND on the corresponding ESL expressions.
4. To save the filter for future use click Create Filter, enter a name, and click OK.
5. Click OK to apply the filter immediately.

2.4.4 Including Entries in the Workspace

In addition to selecting entries, you can also use the Project Table to control which entries are displayed in the Workspace. An entry that is displayed in the Workspace is *included* in the Workspace; likewise, an entry that is not displayed is *excluded*. Included entries are marked by an X in the diamond in the In column; excluded entries are marked by an empty diamond. Entry inclusion is completely independent of entry selection.

To include or exclude entries, click, shift-click, or control-click in the In column of the entries, or select entries and choose Include or Exclude from the Entry menu. Inclusion with the mouse works just like selection: when you include an entry by clicking, all other entries are excluded.

It is sometimes useful to keep one entry in the Workspace and include others one by one: for example, a receptor and a set of ligands. You can fix the receptor in the Workspace by selecting it in the Project Table and choosing Fix from the Entry menu or by pressing CTRL+F. A padlock icon replaces the diamond in the In column to denote a *fixed* entry. To remove a fixed entry from the Workspace, you must exclude it explicitly (CTRL+X). It is not affected by the inclusion or exclusion of other entries. Fixing an entry affects only its inclusion; you can still rotate, translate, or modify the structure.

2.4.5 Mouse Functions in the Project Table

The Project Table supports the standard use of shift-click and control-click to select objects. This behavior applies to the selection of entries and the inclusion of entries in the Workspace. You can also drag to resize rows and columns and to move rows.

You can drag a set of non-contiguous entries to reposition them in the Project Table. When you release the mouse button, the entries are placed after the first unselected entry that precedes the entry on which the cursor is resting. For example, if you select entries 2, 4, and 6, and release the mouse button on entry 3, these three entries are placed after entry 1, because entry 1 is the first unselected entry that precedes entry 3. To move entries to the top of the table, drag them above the top of the table; to move entries to the end of the table, drag them below the end of the table.

A summary of mouse functions in the Project Table is provided in Table 2.3.

Table 2.3. Mouse operations in the Project Table.

Task	Mouse Operation
Change a Boolean property value	Click repeatedly in a cell to cycle through the possible values (On, Off, Clear)
Display the Entry menu for an entry	Right-click anywhere in the entry. If the entry is not selected, it becomes the selected entry. If the entry is selected, the action is applied to all selected entries.
Display a version of the Property menu for a property	Right-click in the column header
Edit the text or the value in a table cell	Click in the cell and edit the text or value
Include an entry in the Workspace, exclude all others	Click the In column of the entry
Move selected entries	Drag the entries
Paste text into a table cell	Middle-click
Resize rows or columns	Drag the boundary with the middle mouse button
Select an entry, deselect all others	For an unselected entry, click anywhere in the row except the In column; for a selected entry, click the row number.
Select or include multiple entries	Click the first entry then shift-click the last entry
Toggle the selection or inclusion state	Control-click the entry or the In column

2.4.6 Project Table Shortcut Keys

Some frequently used project operations have been assigned shortcut key combinations. The shortcuts, their functions, and their menu equivalents are listed in Table 2.4.

Table 2.4. Shortcut keys in the Project Table.

Keys	Action	Equivalent Menu Choices
CTRL+A	Select all entries	Select > All
CTRL+F	Fix entry in Workspace	Entry > Fix
CTRL+I	Open Import panel	Table > Import Structures
CTRL+N	Include only selected entries	Entry > Include Only
CTRL+U	Deselect all entries	Select > None
CTRL+X	Exclude selected entries	Entry > Exclude
CTRL+Z	Undo/Redo last command	Edit > Undo/Redo in main window

2.5 Building a Structure

After you start Maestro, the first task is usually to create or import a structure. You can open existing Maestro projects or import structures from other sources to obtain a structure, or you can build your own. To open the Build panel, do one of the following:

- Click the Open/Close Build panel button in the toolbar:

- Choose Build from the Edit menu.
- Press CTRL+B.

The Build panel allows you to create structures by drawing or placing atoms or fragments in the Workspace and connecting them into a larger structure, to adjust atom positions and bond orders, and to change atom properties. This panel contains a toolbar and three folders.

2.5.1 Placing and Connecting Fragments

The Build panel provides several tools for creating structures in the Workspace. You can place and connect fragments, or you can draw a structure freehand.

To place a fragment in the Workspace:

1. Select Place.
2. Choose a fragment library from the Fragments menu.
3. Click a fragment.
4. Click in the Workspace where you want the fragment to be placed.

To connect fragments in the Workspace, do one of the following:

- Place another fragment and connect them using the Connect & Fuse panel, which you open from the Edit menu on the main menu bar or with the Display Connect & Fuse panel on the Build toolbar.

- Replace one or more atoms in the existing fragment with another fragment by selecting a fragment and clicking in the Workspace on the main atom to be replaced.
- Grow another fragment by selecting Grow in the Build panel and clicking the fragment you want to add in the Fragments folder.

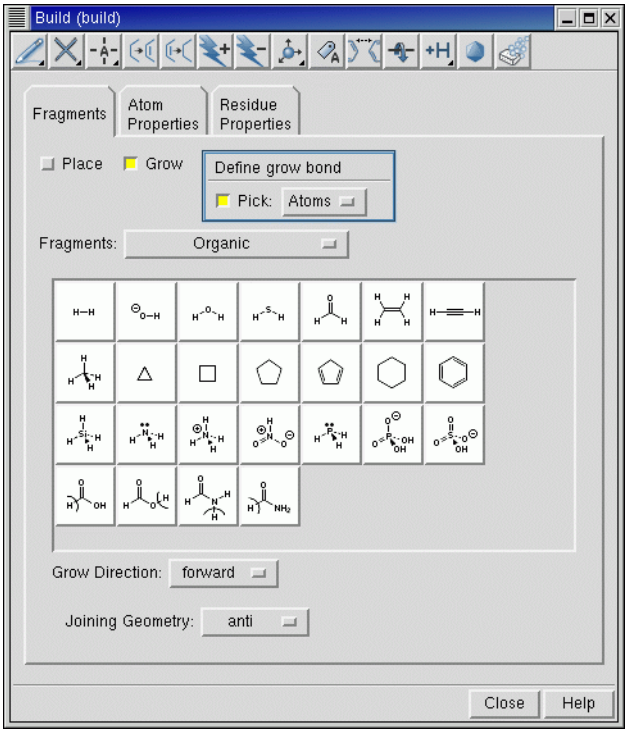


Figure 2.4. The* Build *panel.

Grow mode uses predefined rules to connect a fragment to the *grow bond*. The grow bond is marked by a green arrow. The new fragment replaces the atom at the head of the arrow on the grow bond and all atoms attached to it. To change the grow bond, choose Bonds from the Pick option menu in the Build panel and click on the desired grow bond in the Workspace. The arrow points to the atom nearest to where you clicked.

To draw a structure freehand:

1. Choose an element from the Draw button menu on the Build panel toolbar:

2. Click in the Workspace to place an atom of that element.
3. Click again to place another atom and connect it to the previous atom.
4. Continue this process until you have drawn the structure.
5. Click the active atom again to finish drawing.

2.5.2 Adjusting Properties

In the Atom Properties folder, you can change the properties of the atoms in the Workspace. For each item on the Property option menu—Element, Atom Type (MacroModel), Partial Charge, PDB Atom Name, Grow Name, and Atom Name—there is a set of tools you can use to change the atom properties. For example, the Element tools consist of a periodic table from which you can choose an element and select an atom to change it to an atom of the selected element.

Similarly, the Residue Properties folder provides tools for changing the properties of residues: the Residue Number, the Residue Name, and the Chain Name.

To adjust bond lengths, bond angles, dihedral angles, and chiralities during or after building a structure, use the Adjust distances, angles or dihedrals button on the main toolbar:

You can also open the Adjust panel from this button menu, from the Display Adjust panel button on the Build panel toolbar (which has the same appearance as the above button) or from the Edit menu in the main window.

2.5.3 The Build Panel Toolbar

The toolbar of the Build panel provides quick access to tools for drawing and modifying structures and labeling atoms. See Section 2.3.2 on page 9 for a description of the types of toolbar buttons. The toolbar buttons and their use are described below.

Free-hand drawing
Choose an element for drawing structures freehand in the Workspace (default C). Each click in the Workspace places an atom and connects it to the previous atom.

Delete
Choose an object for deleting. Same as the Delete button on the main toolbar, see page 10.

Set element
Choose an element for changing atoms in the Workspace (default C). Click an atom to change it to the selected element.

Increment bond order
Select a bond to increase its bond order by one, to a maximum of 3.

Decrement bond order
Select a bond to decrease its bond order by one, to a minimum of 0.

Increment formal charge
Select an atom to increase its formal charge by one.

Decrement formal charge
Select an atom to decrease its formal charge by one.

Move
Choose a direction for moving atoms, then click the atom to be moved. Moves in the XY plane are made by clicking the new location. Moves in the Z direction are made in 0.5 Å increments.

Label
Apply heteroatom labels as you build a structure. The label consists of the element name and formal charge, and is applied to atoms other than C and H.

Display Connect & Fuse panel
Open the Connect & Fuse panel so you can connect structures (create bonds between structures) or fuse structures (replace atoms of one structure with those of another).

Display Adjust panel
Open the Adjust panel so you can change bond lengths, bond angles, dihedral angles, or atom chiralities.

Add hydrogens
Choose an atom type for applying the current hydrogen treatment. Same as the Add hydrogens button on the main toolbar, see page 10.

Geometry Symmetrizer
Open the Geometry Symmetrizer panel for symmetrizing the geometry of the structure in the Workspace.

Geometry Cleanup
Clean up the geometry of the structure in the Workspace.

2.6 Selecting Atoms

Maestro has a powerful set of tools for selecting atoms in a structure: toolbar buttons, picking tools in panels, and the Atom Selection dialog box. These tools allow you to select atoms in two ways:

- Select atoms first and apply an action to them
- Choose an action first and then select atoms for that action

2.6.1 Toolbar Buttons

The small triangle in the lower right corner of a toolbar button indicates that the button contains a menu. Many of these buttons allow you to choose an object type for selecting: choose Atoms, Bonds, Residues, Chains, Molecules, or Entries, then click on an atom in the Workspace to perform the action on all the atoms in that structural unit.

For example, to select atoms with the Workspace selection toolbar button:

1. Choose Residues from the Workspace selection button menu:

   The button changes to:

2. Click on an atom in a residue in the Workspace to select all the atoms in that residue.

2.6.2 Picking Tools

The picking tools are embedded in each panel in which you need to select atoms to apply an operation. The picking tools in a panel can include one or more of the following:

- Pick option menu—Allows you to choose an object type. Depending on the operation to be performed, you can choose Atoms, Bonds, Residues, Chains, Molecules, or Entries, then click on an atom in the Workspace to perform the action on all the atoms in that structural unit.

  The Pick option menu varies from panel to panel, because not all object types are appropriate for a given operation. For example, some panels have only Atoms and Bonds in the Pick option menu.

- All button—Performs the action on all atoms in the Workspace.
- Selection button—Performs the action on any atoms already selected in the Workspace.
- Previous button—Performs the action on the most recent atom selection defined in the Atom Selection dialog box.
- Select button—Opens the Atom Selection dialog box.
- ASL text box—Allows you to type in an ASL expression for selecting atoms.

  ASL stands for Atom Specification Language, and is described in detail in the *Maestro Command Reference Manual*.

- Clear button—Clears the current selection

- Show markers option—Marks the selected atoms in the Workspace.

For example, to label atoms with the Label Atoms panel:

1. Choose Atom Labels from the Display menu.
2. In the Composition folder, select Element and Atom Number.
3. In the picking tools section at the top of the panel, you could do one of the following:
   - Click Selection to apply labels to the atoms already selected in the Workspace (from the previous example).
   - Choose Residues from the Pick option menu and click on an atom in a different residue to label all the atoms in that residue.

2.6.3 The Atom Selection Dialog Box

If you wish to select atoms based on more complex criteria, you can use the Atom Selection dialog box. To open this dialog box, choose Select from a button menu or click the Select button in a panel. See Section 5.3 of the *Maestro User Manual* for detailed instructions on how to use the Atom Selection dialog box.

2.7 Scripting in Maestro

Although you can perform nearly all Maestro-supported operations through menus and panels, you can also perform operations using Maestro commands, or compilations of these commands, called *scripts*. Scripts can be used to automate lengthy procedures or repetitive tasks and can be created in several ways. These are summarized below.

2.7.1 Python Scripts

Python is a full-featured scripting language that has been embedded in Maestro to extend its scripting facilities. The Python capabilities within Maestro include access to Maestro functionality for dealing with chemical structures, projects, and Maestro files.

The two main Python commands used in Maestro are:

- `pythonrun`—executes a Python module. (You can also use the alias `pyrun`.) The syntax is:

  `pythonrun` *module*`.`*function*

- `pythonimport`—rereads a Python file so that the next time you use the `pythonrun` command, it uses the updated version of the module. (You can also use the alias `pyimp`.)

From the Maestro Scripts menu you can install, manage, and run Python scripts. For more information on the Scripts menu, see Section 13.1 of the *Maestro User Manual*.

For more information on using Python with Maestro, see *Maestro Scripting with Python*.

2.7.2 Command Scripts

All Maestro commands are logged and displayed in the Command Script Editor panel. This means you can create a command script by performing the operations with the GUI controls, copying the logged commands from the Command History list into the Script text area of the panel, then saving the list of copied commands as a script.

To run an existing command script:

1. Open the Command Script Editor panel from the Edit menu in the main window.
2. Click Open Local and navigate to the directory containing the desired script.
3. Select a script in the Files list and click Open.

   The script is loaded into the Script window of the Command Script Editor panel.
4. Click Run Script.

Command scripts cannot be used for Prime operations.

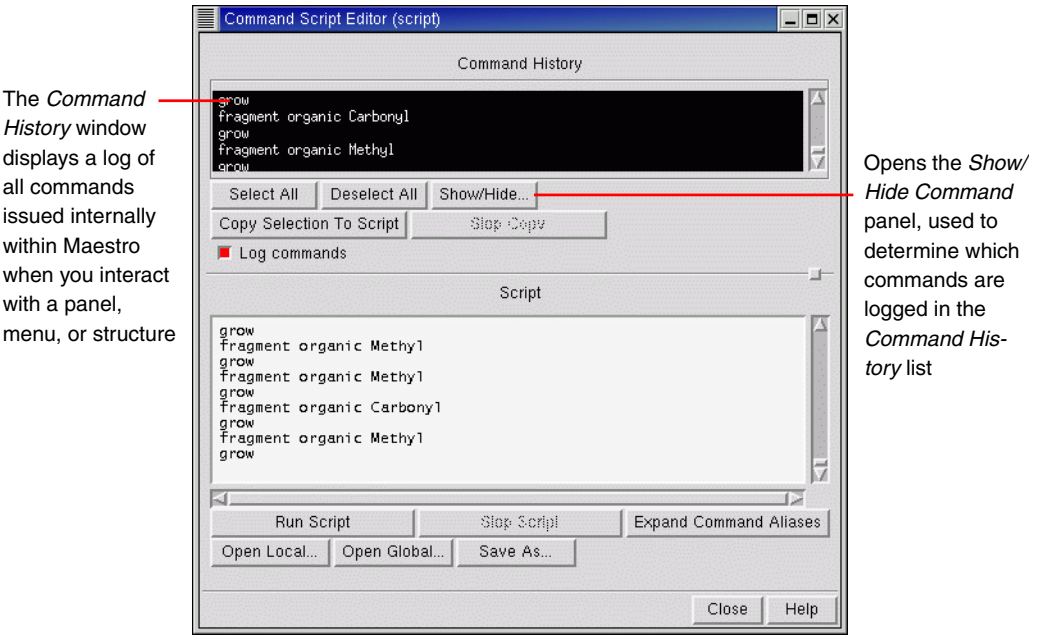


Figure 2.5. The Command Script Editor ***panel.***

2.7.3 Macros

There are two kinds of macros you can create: named macros and macros assigned to function keys F1 through F12.

To create and run a named macro:

1. Open the Macros panel from the Edit menu in the main window.
2. Click New, enter a name for the macro, and click OK.
3. In the Definition text box, type the commands for the macro.
4. Click Update to update the macro definition.
5. To run the macro, enter the following in the command input area in the main window:

   `macrorun` *macro-name*

   If the command input area is not visible, choose Command Input Area from the Display menu.

To create and run a function key macro:

1. Open the Function Key Macros panel from the Edit menu in the main window.
2. From the Macro Key option, select a function key (F1 through F12) to which to assign the macro.
3. In the text box, type the commands for the macro.
4. Click Run to test the macro or click Save to save it.
5. To run the macro from the main window, press the assigned function key.

For more information on macros, see Section 13.5 of the *Maestro User Manual*.

2.8 Specifying a Maestro Working Directory

When you use Maestro to launch Liaison jobs, Maestro writes job output to the directory specified in the Directory folder of the Preferences panel. By default, this directory (the file I/O directory) is the directory from which you started Maestro.

To change the Maestro working directory:

1. Open the Preferences panel from the Maestro menu.
2. Click the Directory tab.
3. Select the directory you want to use for reading and writing files.

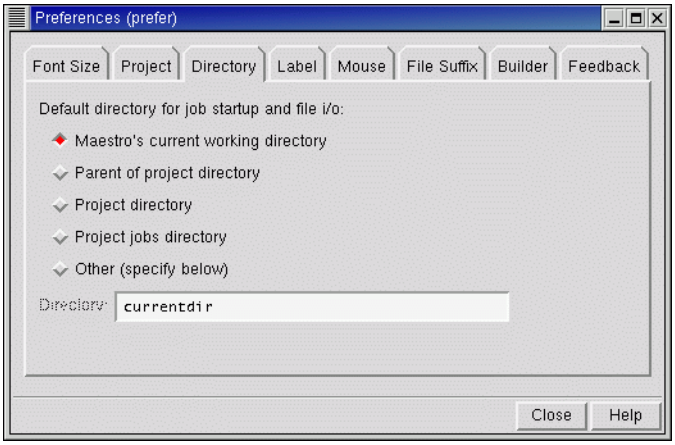


Figure 2.6. T

You can also set other preferences in the Preferences panel. See Section 12.2 of the *Maestro User Manual* for details.

2.9 Undoing an Operation

To undo a single operation, click the Undo button in the toolbar, choose Undo from the Edit menu, or press CTRL+Z. The word Undo in the menu is followed by text that describes the operation to undo. Not all operations can be undone: for example, global rotations and translations are not undoable operations. For such operations you can use the Save view and Restore view buttons in the toolbar, which save and restore a molecular orientation.

2.10 Running and Monitoring Jobs

Maestro has panels for each product for preparing and submitting jobs. To use these panels, choose the appropriate product and task from the Applications menu and its submenus. Set the appropriate options in the panel, then click Start to open the Start dialog box and set options for running the job. For a complete description of the Start dialog box associated with your computational program, see your product's User Manual. When you have finished setting the options, click Start to launch the job and open the Monitor panel.

The Monitor panel is the control panel for monitoring the progress of jobs and for pausing, resuming, or killing jobs. All jobs that belong to your user ID can be displayed in the Monitor panel, whether or not they were started from Maestro. Subjobs are indented under their parent in the job list. The text pane shows various output information from the monitored job, such as the contents of the log file. The Monitor panel opens automatically when you start a job. If it is

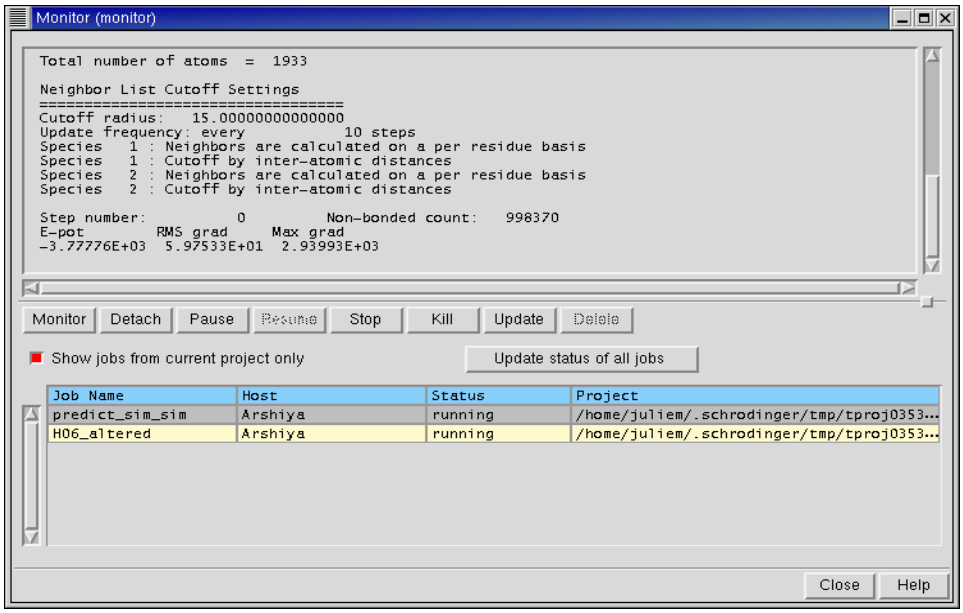


Figure 2.7. The* Monitor *panel.

not open, you can open it by choosing Monitor from the Applications menu in the Maestro main window.

While jobs are running, the Detach, Pause, Resume, Stop, Kill, and Update buttons are active. When there are no jobs currently running, only the Monitor and Delete buttons are active. These buttons act on the selected job. By default, only jobs started from the current project are shown. To show other jobs, deselect Show jobs from current project only.

When a monitored job ends, the results are incorporated into the project according to the settings used to launch the job. If a job that is not currently being monitored ends, you can select it in the Monitor panel and click Monitor to incorporate the results. Monitored jobs are incorporated only if they are part of the current project. You can monitor jobs that are not part of the current project, but their results are not incorporated. To add their results to a project, you must open the project and import the results.

Further information on job control, including configuring your site, monitoring jobs, running jobs, and job incorporation, can be found in the *Job Control Guide* and the *Installation Guide*.

2.11 Getting Help

Maestro comes with automatic, context-sensitive help (Auto-Help), Balloon Help (tooltips), an online help facility, and a user manual. To get help, follow the steps below:

- Check the Auto-Help text box at the bottom of the main window. If help is available for the task you are performing, it is automatically displayed there. It describes what actions are needed to perform the task.
- If your question concerns a GUI element, such as a button or option, there may be Balloon Help for the item. Pause the cursor over the element. If the Balloon Help does not appear, check that Show Balloon Help is selected in the Help menu of the main window. If there is Balloon Help for the element, it appears within a few seconds.
- If you do not find the help you need using either of the steps above, click the Help button in the lower right corner of the appropriate panel. The Help panel is displayed with a relevant help topic.
- For help with a concept or action not associated with a panel, open the Help panel from the Help menu or press CTRL+H.

If you do not find the information you need in the Maestro help system, check the following sources:

- The *Maestro User Manual*
- The Frequently Asked Questions page, found at http://www.schrodinger.com/Support/faq.html

You can also contact Schrödinger by e-mail or phone for help:

- E-mail: help@schrodinger.com
- Phone: (503) 299-1150

2.12 Ending a Maestro Session

To end a Maestro session, choose Quit from the Maestro menu. To save a log file with a record of all operations performed in the current session, click Quit, save log file in the Quit panel. This information can be useful to Schrödinger support staff when responding to any problem you report.

Liaison Tutorial

This chapter contains tutorial exercises to help you quickly become familiar with the functionality of Liaison using the Maestro interface. Liaison is used to simulate and predict binding affinities. It does so by generating for each protein-ligand complex the descriptors necessary to apply the LIA equation, the LiaScore, which is the GlideScore computed over discrete protein atom locations rather than over a grided protein representation, and the LiaScore components which may be used to generate an ELR model. Models for the binding affinity are then created and applied from Liaison-generated descriptors via Strike. Thus, the Liaison process involves two steps, the simulation of binding in Liaison to generate a set of descriptors and the creation and application of binding affinity models from the Liaison descriptors with Strike.

The exercises in this chapter demonstrate:

- How to perform Liaison simulations on multiple ligands
- How to create a validated model for the α, β, and γ coefficients for the LIA equation using Strike
- How to generate and apply the results of Liaison simulations to predict binding affinities for novel ligands
- How to create and apply LiaScore and Extended Linear Response (ELR) models to predict binding affinities

You will use the Liaison panel to set up and run Liaison simulations and then use the Strike panels to create, validate, and apply binding affinity models from Strike.

To do these exercises you must have access to an installed version of Maestro 7.5, Liaison 4.0 and Strike 1.5. For installation instructions, see the *Installation Guide*.

3.1 Preparation

Before you start Maestro and begin the exercises, you must first create a local tutorial directory tree. Some exercises in this tutorial produce files that are needed in subsequent exercises. To allow you to begin at any exercise you choose, the `$SCHRODINGER/impact-v`*version*`/tutorial/liaison` directory contains copies of the relevant input and output files that will be generated as you perform the tutorial.

To create a local directory tree:

1. At a shell prompt, change to a directory in which you have write permissions.
2. Create a local base directory:

   `mkdir` *basedir*

   In the text, this directory is referred to as the base directory.
3. In the base directory, create a soft link to the `$SCHRODINGER/impact-v`*version*`/tutorial/liaison` directory by entering:

   `ln -s $SCHRODINGER/impact-v`*version*`/tutorial/liaison .`

3.2 Starting Maestro

You do not need to start Maestro until you begin an exercise. If you have not started Maestro before, this section contains instructions.

To start Maestro:

1. Set the `SCHRODINGER` environment variable to the directory in which Maestro and Liaison are installed:

   csh/tcsh: `setenv SCHRODINGER` *installation_path*

   sh/bash/ksh: `export SCHRODINGER=`*installation_path*

2. Change to the desired working directory.

   `cd` *basedir*
3. Enter the command:

   `$SCHRODINGER/maestro &`

You are now ready to proceed with the exercises below.

3.3 Running the Liaison Simulations

The Liaison calculations to be run in this exercise require about 1.5 hours of CPU time on a single 2.8 GHz Xeon processor, though by taking advantage of multiple processors this time can be reduced sharply. The output files from the Liaison simulation have been prepared for this exercise so the tutorial may be completed whether you choose to run the Liaison simulations or not.

3.3.1 Setting Job Parameters

Before running Liaison on a series of ligand-receptor complexes, you must import the receptor and ligands and set the parameters for the Liaison simulations.

1. Choose Liaison from the Applications menu in the main window.

   The Liaison panel opens with the Systems folder displayed.

2. In the Job options section, change the Job Name to `hept_analogs`, and select the host where the Liaison simulations are to be run from the Host option menu.

   To run on the local machine ensure that `localhost` is selected. If the host is a multiprocessor machine, specify the number of available CPUs in the Number of Processors text box, otherwise leave its value at one.

3. In the Specify structures section, ensure that Take complexes from a Maestro Pose Viewer File is selected.

4. Click Browse.

5. In the file selector, navigate to your *basedir*`/liaison` directory and import the file `1rt1_hept_analogs_pv.mae`.

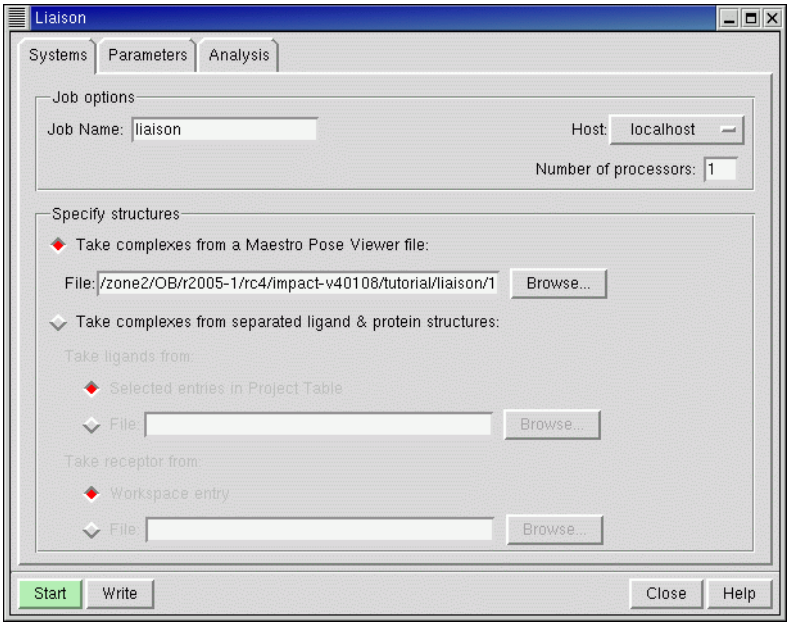


Figure 3.1. The Systems ***folder of the*** Liaison ***panel.***

The ligand-protein complexes to be simulated have now been specified. To prepare the Liaison simulation parameters:

6. In the Job options section of the Parameters folder, choose OPLS_2001 from the Force field option menu.
7. In the Specify restrained/frozen shells section, choose Huge from the option menu.

3.3.2 Starting and Monitoring the Liaison Job

The job takes approximately 1.5 hours on a 2.8 GHz Xeon processor; the time will vary depending on your particular system configuration, workload, and number of processors used for the calculation. The Liaison simulations do not need to be run to continue with the tutorial. Ifyou prefer, you may continue the tutorial starting with Section 3.4.

With the ligands and receptors defined and Liaison simulation parameters set, the Liaison simulation can be started.

1. Click Start on the lower left corner of the Liaison panel.

   A dialog box opens, indicating the job has started and displaying the job ID.

2. Click OK.

   The Monitor panel opens.

While the `hept_analogs` job is in progress, the Status column for this job displays the text "running". When the job is complete, the status changes to "incorporated : finished". The `hept_analogs` job spawns a subprocess named `liaison`. This subprocess in turn spawns a job for each ligand-receptor complex.

This Liaison calculation is being run using the Ligand & Structure-Based Descriptors (LSBD) computational engine. For further details on LSBD, see the online help.

Before the job is launched the following input files are written:

`hept_analogs.inp`	LSBD command input to run Liaison simulation
`hept_anlaogs-ligs.mae`	Ligand structure input for Liaison simulation
`hept_analogs-rec.mae`	Receptor structure input for Liaison simulation
`hept_analogs-lia-cons.mae`	Ligand structure from which restrained/frozen receptor atoms are determined

When the Liaison simulation finishes, the calculated Liaison results are incorporated into the Project Table along with the input ligand geometries, and the *basedir* directory will contain the following job output files:

`hept_analogs.log`	LSBD log summary file
`hept_analogs-final.mae`	Structure file of input ligand geometries with calculated Liaison results
`hept_analogs.csv`	Comma-separated value file with calculated Liaison results
`hept_analogs.tar.gz`	Compressed tarball containing the full set of Liaison results

3.3.3 Exercise Summary

Starting with a receptor and a set of prepared ligands, you have set up and run a job that calculated Liaison descriptors. The descriptors are saved in a Maestro file and a CSV file, allowing you to use them in the creation, validation, and application of LIA, LiaScore and ELR models of binding affinity.

If you want to stop working on the tutorial now, choose Close Project from the Project menu on the main menu bar. If the project is a scratch project, you will be prompted to save it or delete it.

3.4 Generating and Validating an LIA Model of Binding Affinity

Before you begin the exercises shown below, you must first have created a local tutorial directory tree, as described in Section 3.1 on page 31. If you have not yet set up your tutorial directory tree, do so now, then proceed with the exercises.

3.4.1 Importing Liaison Results into Strike for Model Creation

You will be importing the Liaison descriptors that will be used to create the LIA model through the Liaison panel. The data could also be imported directly into the Project Table. The Liaison descriptors are stored in both the `hept_analog-final.mae` and `hept_analog.csv` files. For the purposes of this tutorial you will import the data using the Maestro file, into a new project.

1. Choose New from the Project menu in the main window.

   A project selector is displayed.

2. Add `liasim` to the end of the text in the Project text box, and click Create.

3. Click Open/Close Project Table on the main toolbar.

The Project Table panel is displayed.

4. Choose Liaison from the Applications menu in the main window.

The Liaison panel opens with the Systems folder displayed.

5. In the Analysis folder, click Browse.

A file selector opens.

6. Navigate to and import the file `hept_analogs-final.mae`.

If you ran the Liaison simulation described in Section 3.3, this file is in your *basedir* directory. Otherwise, you can find a copy of it in your *basedir*`/liaison` directory.

7. In the Analysis folder, click Fit or Predict in Strike.

The molecules and data are imported from `hept_analogs-final.mae` into the Project Table, and the Strike Build QSAR Model panel opens.

8. Close the Liaison panel.

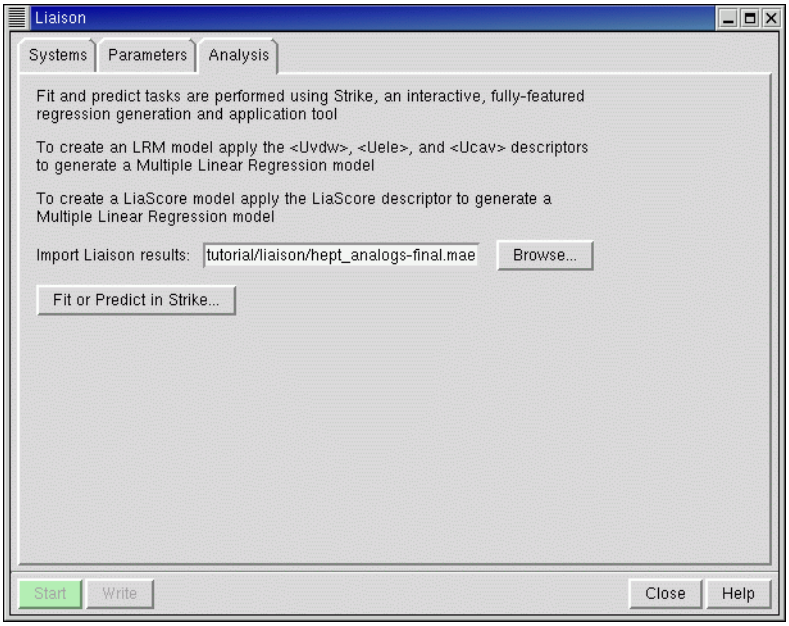


Figure 3.2. The Analysis ***folder of the*** Liaison ***panel.***

3.4.2 Selecting Training Set Molecules

It is often important before model generation to separate available data into training and test sets. The training set is used to train a model to predict binding affinities. The resultant model is then validated by applying it to the prediction of binding affinities for molecules in the test set. The current data set has been separated into training and test sets as indicated in the Set property. Typically this separation is done with the Random option in the Select menu of the Project Table panel.

Strike creates models using selected entries in the Project Table. Since we are to generate an LIA model using only molecules in the training set, its important to ensure only training set molecules are selected in the Project Table:

1. Choose Only from the Select menu in the Project Table panel.

   The Entry Selection panel opens. In the Properties folder there is a list of properties stored in the Project Table.

2. Select Set from the list in the Properties folder.

3. Select Matches and enter `Train` in the text field.

4. Click Add.

   The ESL text box is updated with the chosen matching condition.

5. Click OK.

   The Entry Selection panel closes, and in the Project Table only the molecules in the training set are selected.

3.4.3 Creating the LIA Model to Predict Binding Affinities

The experimental binding affinities given in the property Activity (kcal/mol) will be used for the response or dependent variable from which a linear model will be created. The LIA equation is outlined in detail in the Liaison Manual and it estimates binding affinities using $<U_{ele}>$, $<U_{vdw}>$, and $<U_{cav}>$ terms from Liaison.

In this exercise, you will create an LIA model for the twelve training set molecules you selected in the previous exercise. The Build QSAR Model panel should already be open from the exercise in Section 3.4.1. If not, choose Build QSAR Model from the Strike submenu of the Applications menu in the main window.

1. In the Select descriptors to be included in the model table, select Liaison <Uele>, Liaison <Uvdw>, and Liaison <Ucav>.

Use control-click to select the second and third of these descriptors. These are the terms that are used for an LIA model, and will be the independent variables in the model.

2. From the Regression method option menu, choose Multiple Linear Regression.
3. Ensure that Use all selected is selected.
4. Click Choose, to the right of the Activity Property text box.

   The Choose Activity Property dialog box opens.

5. Select the Activity (kcal/mol) property and click OK.

   This property is the response or dependent variable which a model will be created to predict.

6. Click Start.

   The QSAR/Statistics Start dialog box opens.

7. Enter `lrm_model` as the job name.
8. Click Start to begin the Strike job.

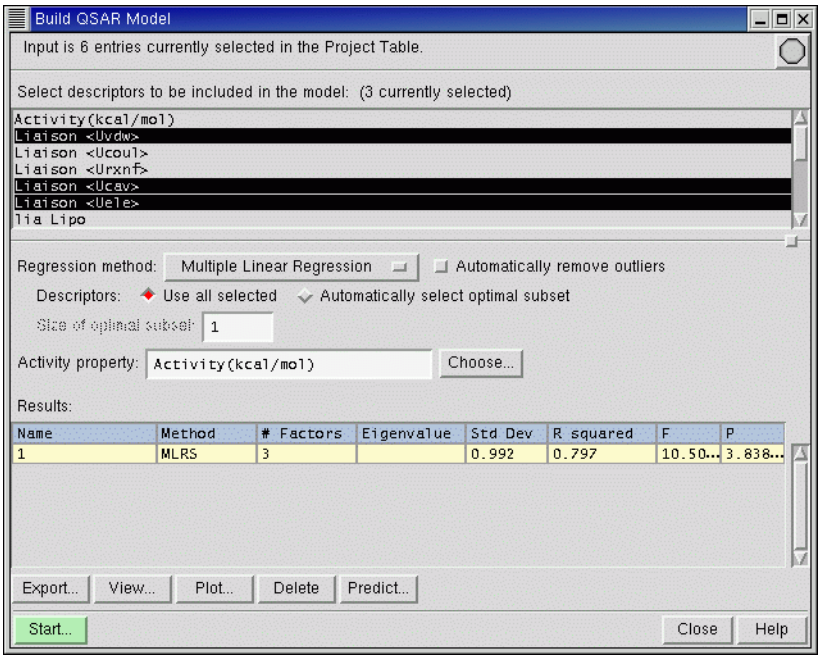


Figure 3.3. The Strike Build QSAR Model ***panel.***

The Strike job should finish in a matter of seconds and the predicted binding affinities are incorporated in the Project Table as Predicted Activity1. In the Build QSAR Model panel, a row with name 1 is added to the Results table that corresponds to the LIA model.

3.4.4 Analyzing the LIA Binding Affinity Model

Once the LIA model has been created, it must be analyzed to see if it makes intuitive sense and possesses predictive power that does not arise by chance. From the Build QSAR Model panel we will view a plot of predicted versus experimental activities, analyze the fundamentals of the model, and assess its predictive power prior to making predictions on test set molecules.

The basics of each model are displayed in the Results table in the Build QSAR Model panel and provide an at-a-glance overview of a model. For a multiple linear regression (MLR) model these include the standard deviation, R^2, F-statistic, and P-value. For the purposes of this tutorial we are interested in models with an R^2 greater than 0.6, a standard deviation lower than 1 log unit, and a P-value less than 0.05. For an LIA model to make intuitive sense the α, β, and γ coefficients calculated when using the OPLS_2001 force-field should all be positive. For OPLS_2005 the cavity term is calculated in a different fashion, and the gamma coefficient does not need to be positive for the LIA model to make intuitive sense. For further information on the fundamental metrics of an MLR model see the *Strike User Manual*.

First, you will display a plot of predicted versus experimental activities.

1. In the Build QSAR Model panel, ensure that the LIA model, which is named model 1, is selected in the Results table.

   Since there is only one model and it is selected already, you should not have to do anything. If there was more than one model, you would have to select the model.

2. Click Plot.

   The Plot XY panel opens with a plot of the predicted binding affinity from the LIA model found in property Predicted Activity1 versus the experimental binding affinities found in property Activity (kcal/mol), which was used as the dependent or response variable in training the MLR model.

   From this plot you can select points and view the selected molecules in the Maestro Workspace, or select groups of molecules in the Project Table. For further information on the full capabilities of the interactive 2D plotting tool see the *Maestro User Manual*.

Next, you will view all the available information on the LIA model.

3. Click View in the Build QSAR Model panel.

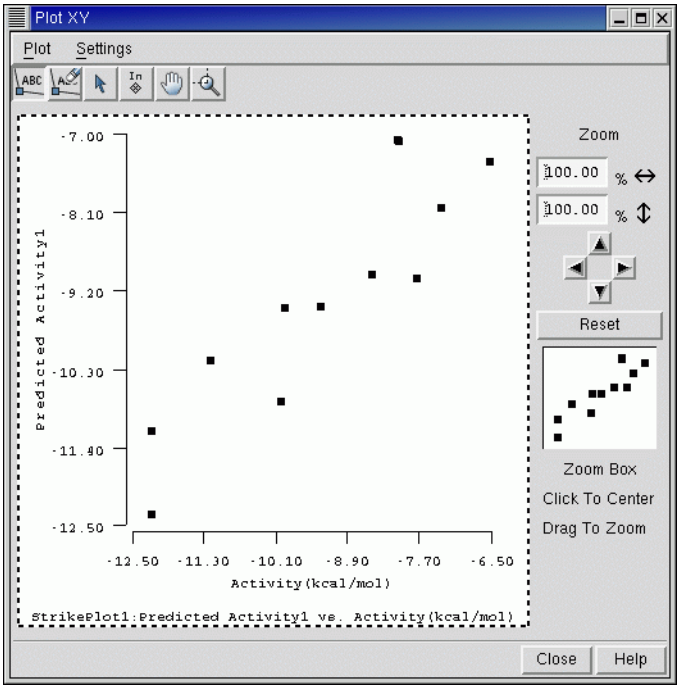


Figure 3.4. The* Plot XY *panel showing the experimental vs predicted activity plot.

The View QSAR Model panel opens. This panel displays the Strike output from LIA model creation. From this display all the information on the model is available, from basic regression statistics to validation tests such as the results of the leave-group-out and dependent variable randomization testing results.

3.4.5 Predicting Binding Affinities with the LIA Model

Once the LIA model has been analyzed and found to be suitable, it may be applied to predict binding affinities for molecules in the test set. In a similar fashion the LIA model may be applied to predict binding affinities for any molecule for which the LIA descriptors have been calculated through Liaison. What is essential is that the molecules must be imported into the Project Table where they can be acted upon by Strike.

First the training set molecules must be selected in the Project Table. If you have changed the selection in the Project Table, repeat the exercise in Section 3.4.2, using `Test` instead of `Train` as the text to match, and skip to Step 2.

1. Choose Invert from the Select menu in the Project Table panel.

   The six training set molecules are selected in the Project Table.

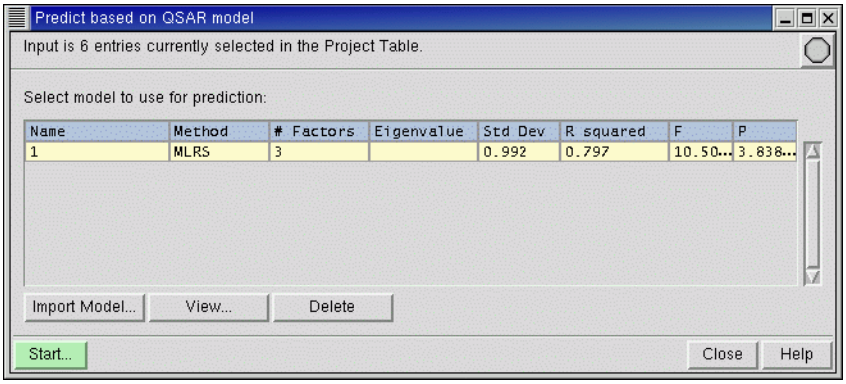


Figure 3.5. The Strike Predict based on QSAR model ***panel.***

2. In the Build QSAR Model panel, click Predict.

   The Predict based on QSAR model panel opens, and the Build QSAR Model panel closes.

3. In the Select model to use for prediction table ensure that the LIA model is selected.

   This should be the model named 1.

4. Click Start.

   The Statistics/Predict - Start dialog box opens.

5. Enter `lrm_pred` as the job name.

6. Choose a host, then click Start to begin the Strike job.

The Strike job should finish in a matter of seconds, and the predicted binding affinities from the LIA model are incorporated into the Project Table as Predicted Activity1.1.

3.4.6 Analyze LIA Binding Affinity Predictions for the Test Set

We are interested in viewing how well the LIA model predicted binding affinities for molecules not included in the training set. Qualitatively the predicted and experimental binding affinities may be plotted using the plotting facilities of Maestro. Quantitatively, the correlation between the predicted and experimental binding affinities is given in the R^2 statistic. In this exercise you will generate an R^2 statistic for the test set molecules showing the correlation between the predicted and experimental binding affinities.

1. Ensure that only the six test set molecules are selected in the Project Table.

   This should be the case from the previous exercise unless you have changed the selection.

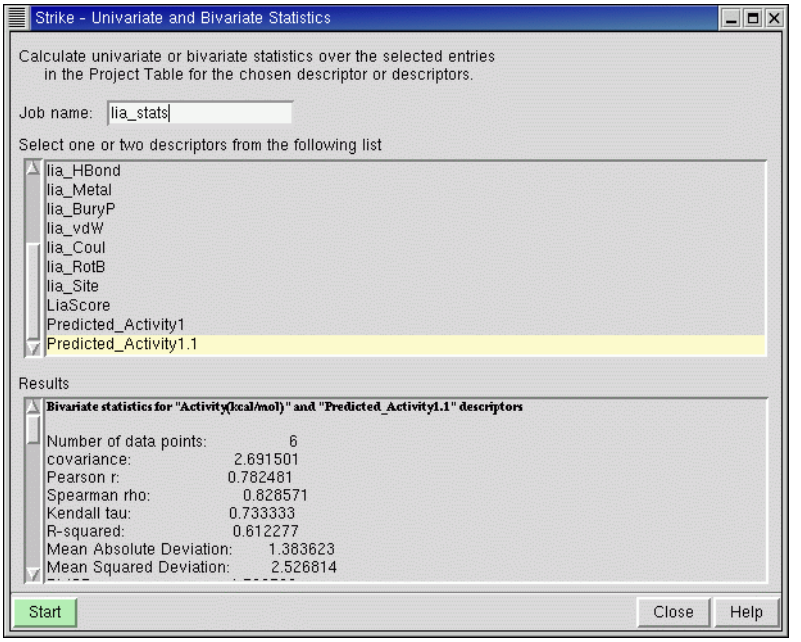


Figure 3.6. The** Strike – Univariate and Bivariate statistics **panel.

2. Choose Statistics from the Strike submenu of the Applications menu in the main window.

   The Strike – Univariate and Bivariate statistics panel opens.

3. Select Activity (kcal/mol) and Predicted Activities1.1 to analyze the correlation between these two properties.

4. Enter `lrm_pred` as the job name.

5. Click Start to begin the calculation.

6. The calculation should take only a few seconds. Once complete the bivariate and univariate statistics are displayed in the Results section, including the R^2 statistic.

3.4.7 Making Predictions for Additional Molecules

Once an LIA model predicting binding affinities has been created and validated, binding affinities may be predicted for any molecule for which the three LIA terms have been generated. The steps are the same as was done for the test set above.

1. Run a Liaison calculation to generate the three LIA descriptors for your own ligands.

2. Import the molecules into the Project Table and ensure that they are selected.

   You can import them using either the Liaison panel or the Import panel.

3. Open the Strike Predict Based on QSAR Model panel.

4. Ensure that the LIA model is selected in the Results table.

5. Click Start.

   The generated predictions are added automatically to the Project Table as Predicted ActivityX.1 where X is the original model number for the LIA model.

3.4.8 Exercise Summary

Starting with Liaison calculated descriptors, you have created, validated, and applied an LIA model to predict binding affinities. If you want to stop working on the tutorial now, choose Close Project from the Project menu in the main menu bar.

3.5 Generating and Applying LiaScore and ELR Binding Affinity Models

In addition to the LIA model, other models of binding affinity may be generated using Liaison calculated properties. From the LiaScore a one-descriptor model may be created. The procedure is the same as was illustrated for the LIA model, though instead of selecting the three Liaison <Uvdw>, Liaison <Uele>, and Liaison <Ucav> terms from the Select descriptors to be included in the model table of the Build QSAR Model panel, simply select the LiaScore descriptor. Model generation, validation, and application then follow exactly as with the LIA model. Sample Strike input and output for the generation and application of the LiaScore only model for the current dataset may be found in:

```
$SCHRODINGER/impact-v version/tutorial/liaison/analysis/ls_*
```

For those interested in expanding the range of descriptors to correlate with binding affinities, an extended linear response (ELR) model may be most appropriate. An example of an ELR model would be to use the LiaScore components, eight descriptors named with a lia prefix, to generate a binding affinity model. With an ELR model you can take advantage of Strike's automatic variable selection or sophisticated partial least squares and principal component linear fitting. Sample input and output for the generation and application of an ELR model using automatic variable selection starting with the eight LiaScore components may be found in:

```
$SCHRODINGER/impact-v version/tutorial/liaison/analysis/liacomp_*
```

Protein Preparation

4.1 Protein and Ligand Structure Preparation

The quality of Liaison results depends on reasonable starting structures. Schrödinger offers a comprehensive protein preparation facility designed to ensure chemical correctness and to optimize protein and protein-ligand complex structures for use as input. It is strongly recommended that protein structures imported from non-Maestro sources, such as PDB structures, be treated with the protein preparation facility in order to achieve best results.

All Liaison and Basic Impact calculations use the OPLS-AA force field. This means that structures must be all-atom (explicit hydrogens) and there must be no covalent bonds between protein atoms, including protein metal atoms, and ligand atoms. Bond orders and formal charges must be correct. The ligand structures for which binding energies are to be predicted must satisfy these requirements as well. For ligand preparation, see Chapter 5.

This chapter describes the preparation of protein-ligand complexes using the Glide protein preparation facility. Most features of the facility are available from the Protein Preparation panel. Additional features are available in the command-line application `protprep`. The utilities `pprep` and `impref` are also available. Use of the command-line application and utilities is summarized in Section 4.11 on page 60.

4.2 The Protein Preparation Panel

The Protein Preparation panel is used to set up jobs for the protein preparation facility, which helps prepare proteins for use in applications such as Liaison, Glide, and QSite. Open the Protein Preparation panel by choosing Protein Preparation from the Glide submenu of the Applications menu in Maestro. A typical PDB structure file consists only of heavy atoms; therefore, hydrogens have to be added prior to use in Liaison calculations, which use an all-atom force field. The charge state of protein residues is also important to the results generated.

The protein preparation facility consists of two components, *preparation* and *refinement*. After ensuring chemical correctness, the *preparation* component neutralizes side chains that are not close to the binding cavity and do not participate in salt bridges. The *refinement* component performs a restrained Impact minimization of the cocrystallized complex, which reorients side-chain hydroxyl groups and alleviates potential steric clashes. The Protein Preparation panel allows you to specify which parts of the procedure to run: Preparation and refinement, Preparation only, or Refinement only. The complete two-component process adds hydrogens, neutral-

izes appropriate amino acid chains, reorients side-chain hydroxyl and thiol groups, and relieves steric clashes.

If you are familiar with Maestro, or if you have followed Schrödinger's protein preparation procedure before, you might need to follow only the overview provided in Section 4.3. The steps are described in detail in later sections of this chapter. For a tutorial on protein preparation, see Chapter 3 of the *Glide Quick Start Guide*.

4.3 Step-by-Step Overview

This section provides an overview of the protein preparation process. The procedure described assumes that the initial protein structure is in a PDB-format file, includes a cocrystallized ligand, and does not include explicit hydrogens. For best results, structures with missing residues near the active site should be repaired before protein preparation. After processing with Schrödinger's protein preparation facility, you will have files containing refined, hydrogenated structures of the ligand and the ligand-receptor complex. The prepared structures are suitable for use with Liaison. In most cases, not all of the steps outlined need to be performed. See the descriptions of each step to determine whether it is required.

1. Import a ligand/protein cocrystallized structure, typically from PDB, into Maestro. The preparation component of the protein preparation facility requires an identified ligand.
2. Locate any waters you want to keep, then delete all others.

   These waters are identified by the oxygen atom, and usually do not have hydrogens attached. Generally, all waters (except those coordinated to metals) are deleted, but waters that bridge between the ligand and the protein are sometimes retained. If any waters are kept, hydrogens will be added to them by the preparation component of the protein preparation job. Afterwards, it is useful to check that these water molecules are correctly oriented.
3. Simplify multimeric complexes.
   - Determine whether the protein-ligand complex is a dimer or other multimer containing duplicate binding sites and duplicate chains that are redundant.
   - If the structure is a multimer with duplicate binding sites, remove redundant binding sites and the associated duplicate chains by picking and deleting molecules or chains in Maestro.
4. Adjust the protein, metal ions, and cofactors.
   - Fix any serious errors in the protein. Incomplete residues are the most common errors, but are relatively harmless if they are distant from the active site. Structures that are missing residues near the active site should be repaired.

- Check the protein structure for metal ions and cofactors.
- Set charges and correct atom types for any metal atoms, as needed.
- Set bond orders and formal charges for any cofactors, as needed.

5. Adjust the ligand bond orders and formal charges.

   If you are working with a dimeric or large protein and two ligands exist in two active sites, the bond orders have to be corrected in both ligand structures.

6. Run protein preparation.

   - Open the Protein Preparation panel, define the ligand by picking, select the desired Procedure, set other options as necessary, and click Start.
   - In the Protein Preparation Start dialog box, type the name of your job in the Name text box, and click Start.

7. Review the prepared structures.

   - If problems arise during the preparation or refinement stages, review the log file, correct the problems, and rerun.
   - Examine the refined ligand/protein/water structure for correct formal charges and protonation states resulting from Step 6 and make final adjustments as needed.

4.4 Importing the Protein Complex Structure

This step begins the protein preparation procedure.

To import a ligand-receptor protein complex structure into Maestro:

1. On the toolbar, click the Import structures button:

   The Import panel is displayed.

2. Select the appropriate format (usually PDB format).
3. Enter the name of the file, or select the file in the Files list.
4. Click Import.
5. To display the Project Table, click the Open/Close project table button on the toolbar:

The imported entry is highlighted in the Project Table and displayed in the Workspace.

4.5 Deleting Unwanted Waters

Water molecules in the crystallographic complex are generally not used unless they are judged critical to the functioning of the protein–ligand interaction. When waters are used, they are later included in the protein as "structural" waters. Keeping structural waters is more likely to be important for Liaison than for other programs such as Glide, where making a site more accessible by removing all waters may be necessary for docking.

4.5.1 Deleting All Water Molecules

- You can delete all waters by choosing Waters from the Delete button menu:

4.5.2 Deleting Distant Water Molecules

If you want to keep one or more waters, you can begin by removing those that are farther than a given distance from the ligand.

1. Choose Select from the Delete button menu.

   The Atom Selection dialog box opens.

2. In the Molecule folder, choose Molecule Number and enter the ligand's molecule number. Click Add.
3. Click the Proximity button in the lower section of the dialog box.
4. In the Proximity dialog box:
   a. Select Beyond, enter a distance in the text box, and select Angstroms.
   b. Under Fill, select Residues and Exclude source.

      This keeps the ligand itself from being deleted.

   c. Click OK to exit the Proximity dialog box.
5. In the Residue folder, choose Residue Type, and select HOH.
6. Click Intersect.

The ASL box should contain an expression similar to:

```
(not (mol.num 2) and fillres beyond 5 (mol.num 2 ) ) AND
((res.ptype "HOH "))
```

and most of the water oxygens are marked in the Workspace.

7. Click OK to delete the selected water molecules.

This task can also be performed using the Commands text box in the lowest part of the Maestro main window by entering a command such as:

`delete res. HOH and beyond` *dist* `mol.n` *molnumlig*

where *molnumlig* is the molecule number of the ligand.

4.5.3 Deleting Remaining Unwanted Waters

After deleting water molecules beyond a certain distance from the ligand, examine the Workspace and delete any remaining water molecules you do not want to keep:

1. On the toolbar, choose Molecules from the Delete button menu:

2. Click on an oxygen to delete that water molecule.

4.6 Simplifying the Protein Complex

4.6.1 Determining Whether the Complex Is a Multimer

To determine whether the ligand-receptor complex is a multimer, compare the chains that appear in the sequence viewer. If there are two or more chains with identical sequences, the complex may be a multimer. If this is the case, there may be duplicate copies of the binding site of interest, with duplicate chains forming the duplicate binding sites.

If the binding interaction of interest takes place within a single subunit, you should retain only one ligand-receptor subunit to prepare for Liaison. However, if two identical chains are both required to form the active site, neither should be deleted. To see whether two duplicate chains are involved with the active site, undisplay the protein's amino acid residues:

1. Choose Protein Backbone from the Undisplay toolbar button menu:

2. Repeat the process and choose Protein Side Chains.

Ligands, cofactors, metal ions, and water oxygens remain visible. If two or more identical ligands or ligand/cofactor groups are present, then the complex is most likely a multimer, and the redundant groups and the duplicate chains associated with them can be deleted.

4.6.2 Retaining Needed Subunits

If the protein complex structure is a multimer with duplicate binding sites, it can be truncated by deleting all but a single ligand binding site and the associated receptor subunit(s). If you choose not to truncate the structure, skip to Section 4.7 on page 51.

To remove redundant subunits or receptor sites of a multimer:

1. Delete all but one ligand or ligand/cofactor pairing:
   a. Choose Molecules from the Delete button menu:

   b. Click on any atom in a molecule to delete that molecule.
2. Display the ligand or ligand/cofactor pair in CPK:
   a. Choose Molecules from the Draw atoms in CPK button menu:

   b. Click on an atom in the ligand to display it in CPK.
   c. If there is a cofactor, click on an atom in that molecule as well.
   d. Click the toolbar button a second time to leave the Draw atoms in CPK pick state.
3. Redisplay the protein backbone:

   Choose Protein Backbone from the Also display toolbar button menu:

   Making just the backbone visible provides enough information without unduly cluttering the Workspace.
4. Assign coloring by Chain Name:

On the toolbar, choose Chain Name from the Color all atoms by scheme button menu:

5. Delete duplicate protein chains:
   a. On the toolbar, choose Chains from the Delete button menu.
   b. Click on a backbone atom in each protein chain you want to delete.
6. Delete duplicate ligands and cofactors:
   a. On the toolbar, choose Molecules from the Delete button menu.
   b. Click on an atom in each ligand or cofactor to be deleted.
7. When finished, redisplay the rest of the protein:

   On the toolbar, choose All from the Display only button menu:

8. Put all atoms, including the ligand and any cofactors, back into wire frame:

   On the toolbar, double-click the Draw bonds in wire button:

4.7 Adjusting the Protein, Metal Ions, and Cofactors

Problems in the PDB protein structure may need to be repaired before it can be used. Incomplete residues are the most common errors, but are relatively harmless if they are distant from the active site. Structures that are missing residues near the active site should be repaired.

If the protein already includes hydrogen atoms, you will need to decide how to proceed. If all hydrogens are present, you could use the structure as is and omit running the protein preparation procedure. This approach is not recommended unless you are absolutely satisfied that the structure is properly prepared and contains no untenable steric clashes.

Typically, you will need to perform the tasks in this section to assure that the protein structure is ready to be prepared. Many of the tools required are in the Build panel, so you should open it before proceeding, by clicking the the Open/Close Build panel button:

4.7.1 Displaying Any Metal Ions and Cofactors

1. Ensure that the entire structure, including any metals and cofactors, is included in the Workspace.
2. On the toolbar, choose Protein Backbone from the Undisplay button menu:

3. Repeat the process and choose Protein Side Chains.
4. As needed, redisplay protein residues near a metal or cofactor, using the Also display button menu and the Atom Selection dialog box.

5. On the Build panel toolbar, click the Label button:

All metal ions (and other heteroatoms) are labeled with their element symbol and formal charge.

6. Examine the protein structure to determine how to continue.
   - If there are formal bonds from protein atoms to metal ions, see Section 4.7.2. Then correct the metal ions as in Section 4.7.3.
   - If the protein contains metal ions without bonds, treat them as described in Section 4.7.3.
   - If the protein contains cofactors, treat them as described in Section 4.7.4.
   - Bonds to the ligand will be discussed in Section 4.8.

4.7.2 Deleting Protein-Metal Bonds

Impact does not permit covalent bonds between metals and protein atoms. The preparation stage of protein preparation automatically deletes protein-metal bonds before hydrogens are added, but this can leave incorrect formal charges on the atoms that were bonded to metal, causing incorrect structures to be generated by protein preparation.

It is safest, therefore, to delete the bonds, then check and adjust element names and formal charges for both metal and non-metal atoms whose bond order you have changed. When you have run protein preparation, it is highly recommended that you examine the prepared complex

for correct protonation states and charges in the active site, then make manual corrections if needed. Before running a Liaison job, you can run a score-in-place Glide calculation on the complex and check that the metal-ligation energy is reasonable. If it is highly positive, you may have to re-adjust the charge and protonation states in the active site manually.

To manually delete bonds between metals and protein atoms:

1. On the toolbar, choose Bonds from the Delete button menu.

2. Click on the bonds to be deleted.
3. You will need to adjust the formal charges of any atoms, metal or otherwise, from which you have deleted a bond. See Section 4.7.3 for metals and Section 4.7.4 for other atoms.

For more information about structure editing in Maestro, click Help or see Chapter 4 of the *Maestro User Manual*.

4.7.3 Adjusting Metal Ion Charges

The MacroModel atom types for metal ions are sometimes incorrectly translated into dummy atom types (Du, Z0, or 00) when metal-protein bonds are specified in the input structure. Furthermore, isolated metal ions may erroneously be assigned general atom types (GA, GB, GC, etc.). The protein preparation procedure cannot treat structure files containing these atom types; they should be corrected as described in this section.

To display element labels and formal charges:

1. On the Build panel toolbar, click the Label button:

   All metal ions (and other heteroatoms) are labeled with their element symbol and formal charge.

2. Check any metal ions to make sure they are correct. If they are, the next step in the process is Section 4.7.4. If not, you can correct them.

To correct metal ion atom types:

1. In the Atom Properties folder of the Build panel, select Atom Type (MacroModel) from the Property option menu.

2. Find the correct atom type for the metal ion.

   The atom type for metal ions includes both element name and formal charge. Atom type numbers are in parentheses.

3. Click in the list to select the correct atom type.
4. Click on the metal ion to be changed.

4.7.4 Displaying and Adjusting the Cofactor

Cofactors are included as part of the protein, but because they are not standard residues it is sometimes necessary to use Maestro's structure-editing capabilities to ensure that multiple bonds and formal charges are assigned correctly.

To display only the cofactor:

1. On the toolbar, choose Select from the Display only button menu:

   The Atom Selection dialog box is displayed.

2. In the Residue folder, choose Residue Type.
3. Click the residue type of the cofactor, which will be near the end of the list.

   The cofactor is highlighted.

4. Click Add, then click OK.

The cofactor is displayed. Because the cofactor was chosen by residue type and not molecule number, this method works even if the cofactor is covalently bonded to another residue.

To set or change cofactor bond orders:

1. On the Build panel toolbar, click the Decrement bond order or Increment bond order button, as appropriate:

2. Click on bonds as necessary to set the bond order.

To set or correct the formal charge on any cofactor atoms:

1. In the Build panel toolbar, click the Label button:

   All metal ions (and other heteroatoms) are labeled with their element symbol and formal charge.

2. On the Build panel toolbar, click on the Increment formal charge or Decrement formal charge button, as appropriate:

3. Click on an atom whose formal charge must be increased or decreased. Repeat as necessary. The atom labels will show the current formal charge.

To correct the atom type of any mis-typed atoms:

1. In the Atom Properties folder of the Build panel, choose Atom Type (MacroModel) from the Property option menu.
2. Find the correct atom type for the mis-typed atom, and click it in the list.
3. Click on the atom to be changed.
4. If the cofactor contains any metal ions, bonds between the metal and cofactor can be removed as in Section 4.7.2.

4.8 Adjusting the Ligand

4.8.1 Manually Deleting Protein-Ligand Bonds

The use of Impact in protein preparation (restrained minimization during refinement) prohibits the existence of formal protein-ligand bonds. This category includes bonds between a protein metal atom and a ligand or cofactor. The OPLS-AA force field models such interactions as a van der Waals plus electrostatic interaction, and the presence of explicit covalent bonds to the ligand, as might be found in an acyl enzyme, interferes with this model. Therefore, Maestro deletes all bonds between the ligand and metal atoms before the protein preparation job begins.

However, setting up the protein preparation job requires the identification of the ligand molecule, and this identification is disrupted by the existence of covalent bonds between protein atoms (metals or otherwise) and ligand atoms. To avoid this, delete such bonds manually.

1. To check for protein-ligand bonds, including protein metal-ligand bonds, use Maestro's Display/Undisplay facility to undisplay the protein backbone and protein side chains. The ligand and any cofactors and metals are displayed.
2. If any metal-ligand or other protein-ligand bonds exist, delete them:
   a. Choose Bonds from the Delete button menu on the toolbar.
   b. Click on the bonds to be deleted.
3. Redisplay the complete protein by choosing All from the Display Only button menu on the toolbar.

4.8.2 Adjusting Ligand Atom and Bond Properties

You will need to adjust the formal charges of any atoms, metal or otherwise, from which you have deleted a bond. Follow the procedures for adjusting metal and cofactor atom and bond properties, as described in Section 4.7.2 and Section 4.7.4.

4.9 Running Protein Preparation on the Structures

From this point on, all structural manipulations are done by the Protein Preparation panel, shown in Figure 4.1, and its related scripts. Before you open this panel, ensure that the protein and ligand are in the Workspace.

To open the Protein Preparation panel, choose Protein Preparation from the Glide submenu of the Applications menu on the main menu bar.

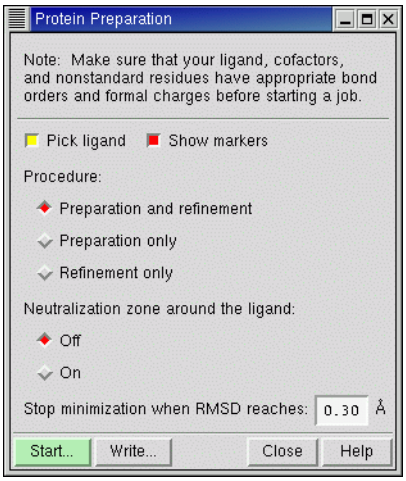


Figure 4.1. The** Protein Preparation **panel.

4.9.1 Defining the Ligand

Before launching a protein preparation job, you must choose a molecule in the Workspace that will be treated as the ligand. In the Protein Preparation panel, select Pick ligand, and then select the ligand by clicking on it in the Workspace. When Show markers is selected, the ligand is highlighted with a blue-green marker. The rest of the Workspace is then treated as the protein.

4.9.2 Choosing a Procedure

The Protein Preparation panel facilitates three types of jobs: Preparation only, Refinement only, and Preparation and refinement.

The Preparation component neutralizes residues that are beyond a set distance from the ligand. The Preparation process also detects some conflicts in hydrogen bonding. It corrects them when possible, either by exchanging carbonyl and hydroxyl oxygens in a neutralized carboxylic acid group, or by creating the alternate (HIE) tautomer of a histidine side chain.

The Refinement component uses Impact to run a series of restrained, partial minimizations on the combined, hydrogenated structure. Minimizations continue until the average RMS deviation of the non-hydrogen atoms reaches the specified limit (0.3 Å by default).

The first step in the sequence of restrained minimizations reorients side-chain hydroxyls in serine, threonine, and tyrosine residues, and side-chain sulfhydryls of cysteine residues. This is accomplished by tightly tethering non-hydrogen atoms with a force constant of 10 kcal/mol·Å^2 and by minimizing the hydrogens with torsion interactions turned off.

Each restrained minimization employs a limited number of minimization steps and is not intended to minimize the system completely. Subsequent steps restore the torsion potential and use progressively weaker restraints on the non-hydrogen atoms (hydrogen atoms are always free). The force constants employed are 3, 1, 0.3, and 0.1 kcal/mol·Å^2.

Preparation and refinement, the default, runs both components. This is the recommended mode if you have not yet run any preparation jobs on the protein. Separate Preparation only and Refinement only jobs can be run if you encountered a problem in the combined Preparation and refinement job. Subsequent Refinement only jobs can be performed after a Preparation and refinement job if water molecules need to be reoriented or if other structural adjustments need to be made.

4.9.3 Other Options

1. Neutralization zone around the ligand

   You can turn on or off a neutralization zone extending to 10-20 Å from the ligand. Residues inside this zone are neutralized.

2. Stop minimization when RMSD reaches: 0.30 Å

   This is the default value. It allows the refinement portion of the job to halt when the average RMSD of the heavy atoms reaches 0.30 Å.

4.9.4 Starting the Protein Preparation Job

To start the Protein Preparation job, click Start. The Start dialog box is displayed, in which you can select a job name and a host, and enter your user name on the host if it is different from the host on which Maestro is running.

To launch a Protein Preparation job, click Start. The monitor panel is displayed, and the results of the job are shown. If you decide to run preparation and refinement separately, you will need to run a Preparation only job and refine the results with a subsequent Refinement only job. If you want to run the job from the command line, click Write Files. See Section 4.11 on page 60 for information on command line options.

When the job finishes, the results are appended to the Project Table. The protein and the ligand are incorporated as separate entries.

4.9.5 Output Job Files

Running Preparation and refinement produces the same files as running Preparation only followed by Refinement only. The following structure files are produced, where *struct* is the name of the complex:

*struct*_lig.mae	The input ligand structure file
*struct*_lig_prep.mae	The post-preparation ligand structure file
*struct*_lig_ref.mae	The post-refinement ligand structure file
*struct*_prot.mae	The input receptor structure file
*struct*_prot_prep.mae	The post-preparation receptor structure file
*struct*_prot_ref.mae	The post-refinement receptor and ligand structures
struct.log	The log file for the complete preparation and refinement job

4.10 Checking the Output Structures

Finally, after both the preparation and refinement components have successfully run, you should check the completed ligand and protein structures.

4.10.1 Checking the Orientation of Water Molecules

Perform this step only if you identified and kept some structural waters in Section 4.5. Reorienting the hydrogens is not strictly necessary, as their orientation should have been changed during refinement in Section 4.9, but it is useful to check that the orientation is correct.

If the orientation is incorrect, reorient the molecules by using the following procedure:

1. Choose Global/Local from the Local transformation toolbar button menu:

   The Advanced Transformations panel is displayed.
2. Under Atoms For Transformation, use the picking controls to select the entire water molecule you want to reorient.
3. Under A Center For Transformation, use the picking controls to select the oxygen atom of the water molecule.
4. Under Rotation/Translation Scope, select Local.
5. Use the middle mouse button to change the orientation of the water hydrogens.
6. Close the Advanced Transformations panel. Transformations should now be global again.

When you have corrected the orientation of the retained water molecules, run a Refinement only job on the adjusted protein-ligand complex as described in Section 4.9.

4.10.2 Resolving H-Bonding Conflicts

One or more residues may need to be modified to resolve an acceptor-acceptor or donor-donor clash. If residues need to be modified, follow these steps:

1. Place the refined protein-ligand complex in the Workspace.
2. Examine the interaction between the ligand and the protein (and/or the cofactor).
3. Use your judgment and chemical intuition to determine which protonation state and tautomeric form the residues in question should have.
4. Use the structure-editing capabilities in Maestro to resolve the conflict.

Some of these clashes are recognized by the preparation process but cannot be resolved by it. The preparation process may have no control over other clashes. An example of the latter typically occurs in an aspartyl protease such as HIV, where both active-site aspartates are close to one or more atoms of a properly docked ligand. Because these contact distances fall within any

reasonable cavity radius, the carboxylates are not subject to being neutralized and will both be represented as negatively charged by the preparation process. However, when the ligand interacts with the aspartates via a hydroxyl group or similar neutral functionality, one of the aspartates is typically modeled as neutral.

4.11 Command-Line Protein Preparation

To run protein preparation from the command line:

1. If you do not yet have receptor and ligand structure files for the structures in the Workspace, use the Write button in the Protein Preparation panel to write the structure files.
2. Use the `protprep`, `pprep`, or `impref` command-line utilities to run specific procedures. These commands and their options are summarized below.

4.11.1 Usage Summary for `protprep`

The `$SCHRODINGER/protprep` application has command-line options corresponding to features of the Maestro Protein Preparation panel.

Syntax:

```
$SCHRODINGER/protprep [options] input-file
```

input-file is the file containing the protein to be prepared or refined. This file must be in Maestro format. When doing a refinement only job (`-mode refine`) this file can contain a protein-ligand complex.

Options:

General Options:

`-j` *jobname*	Override the default job name derived from *input-file*. This allows you to choose an output job name that is different from the *input-file* name.
`-l` *ligand-file*	Specify a file containing a ligand in the protein's active site. This file must be in Maestro format.
`-m` *mode* `-mode` *mode*	Select mode, where *mode* is one of the following: `prep` Preparation only. `refine` Refinement only. `both` Preparation and refinement (default).
`-debug`	Print verbose (debugging) output.

`-HOST` *host*	Run the job on a remote host.
`-LOCAL`	Run the job in the current directory, rather than in a temporary scratch directory.
`-WAIT`	Wait until the job finishes before returning the command prompt.
`-NICE`	Run the job at reduced priority.
`-HELP | -h`	Print usage message and exit.

Preparation Stage Options:

`-min-recep-only`	Minimize total charge of receptor only.
`-skip-sidechain-corr`	Skip correction of conflicting side-chain forms.
`-cavity-8-12`	Set cavity distance range to 8-12 Å. Suitable for Liaison jobs.
`-salt-bridge-dist`	Leave residue pairs forming salt bridges within *dist* Å ionized; default is 3.5 Å.
`-ionization-range`	Leave residues within *dist* Å of ligand ionized.
`-hbond-dist`	Set H-bonding distance; default is 3.45 Å.

Refinement Stage Options:

`-r` *rmsd*	Maximum RMSD allowed for refinement; default 0.3.
`-keep`	Keep intermediate structure files.
`-separate`	Write out refined protein and ligand structures separately, rather than in one combined structure.

4.11.2 Usage Summary for `pprep`

The purpose of `pprep` is to adjust protonation states of a receptor in a Maestro format file. `pprep` is the driver for the preparation stage, and is called by `protprep`. There should be little need to run `pprep` directly.

Syntax:

```
$SCHRODINGER/utilities/pprep [options] proteinfile.mae
```

Options:

`-i` *idis*	Leave residue pairs forming salt bridges within *idis* ionized; default is 3.5 Å.
`-l` *ligfile*	Read ligand mae file *ligfile*.
`-n` *outfile*	Specify non-default (*lig*`R.mae`) name for output file with neutralized residues.
`-p`	Print verbose output.
`-r`	Minimize total charge of the receptor only.
`-t`	Skip correction of conflicting side-chain forms.
`-w` *wdis*	Leave residues within *wdis* of ligand ionized.
`-H` *hbonddist*	Set H-bonding distance; default 3.45 Å.
`-L`	Set cavity distance range to 8-12 Å.
`-v`	Print version number and exit.
`-h`	Print usage message and exit.

4.11.3 Usage Summary for `impref`

The purpose of `impref` is to use Impact for restrained optimizations of a ligand-receptor complex. `impref` is the driver for the refinement stage, and is invoked by `protprep`. There is little need to run `impref` directly.

Syntax:

```
$SCHRODINGER/utilities/impref [options] input.mae
```

Options:

`-k`	Keep Impact minimization *`.inp`, *`.log`, and *`.mae` files.
`-l` *ligfile*	Read ligand from file *ligfile*, instead of *input*`.mae`. If this option is used, *input*`.mae` must be the protein structure alone. If this option is not used, *input*`.mae` must be the protein/ligand complex.
`-r` *rmsd*	Specify maximum RMSD allowed; default is 0.3.
`-s`	Write out protein and ligand separately. Requires `-l` *ligfile*.
`-op` *file*	Output protein or complex file. Default is *input*`_ref.mae`.
`-ol` *file*	Output ligand file (when `-s` and `-l` used.) Default is *ligfile*`_ref.mae`.
`-v`	Print version number and exit.
`-h`	Print usage message and exit.

Ligand Preparation

5.1 Ligand Preparation Checklist

Chapter 4 discussed the preparation of receptor and ligand-receptor structure files for use in Liaison. Ligand structures must also have certain characteristics for Liaison. Some of these conditions can be met by using Maestro features or command-line utilities to alter the ligand structure file.

To be submitted to Liaison or Impact applications, ligand structures:

1. Must be three-dimensional (3D).
2. Must each consist of a single molecule that has no covalent bonds to the receptor, with no accompanying fragments, such as counter-ions and solvent molecules.
3. Must be in a Maestro-format file. Maestro transparently converts SD, MacroModel, and PDB formats to Maestro format during structure import. Maestro also transparently converts Mol2 during import. However, Liaison has no direct Mol2 support, so ensure your structures are in Maestro, SD, or PDB format before starting Liaison jobs.

   Structure file format conversion can be done from the command line with utilities such as `pdbconvert`, `sdconvert`, and `maemmod`—see Appendix D of the *Maestro User Manual*.
4. Must have all their hydrogens (filled valences). These can be added in Maestro by using either the Add hydrogens toolbar button:

   or the Hydrogen Treatment panel (select Hydrogen Treatment from the Edit menu).

   Hydrogen atoms can also be added (or removed) using the command-line tool `applyhtreat`, which is described in Appendix D of the *Maestro User Manual*.

5.2 LigPrep

The Schrödinger ligand preparation product LigPrep is designed to prepare high quality, all-atom 3D structures for large numbers of drug-like molecules, starting with 2D or 3D structures in SD or Maestro format. LigPrep can be run from Maestro or from the command line. For detailed information on LigPrep, see the *LigPrep User Manual*.

To run LigPrep, you must have a LigPrep license. The MacroModel commands `premin` and `bmin` require LigPrep licenses when run in a LigPrep context, and are limited to a restricted set of commands when run using a LigPrep license. For more information about obtaining LigPrep, contact help@schrodinger.com.

The simplest use of LigPrep produces a single low-energy 3D structure with correct chiralities for each successfully processed input structure. LigPrep can also produce a number of structures from each input structure with various ionization states, tautomers, stereochemistries, and ring conformations, and eliminate molecules using various criteria including molecular weight or specified numbers and types of functional groups present.

5.2.1 The LigPrep Process

The LigPrep process consists of a series of steps that perform conversions, apply corrections to the structures, generate variations on the structures, eliminate unwanted structures, and optimize the structures. Many of the steps are optional, and are controlled by selecting options in the LigPrep panel or specifying command-line options. The steps are outlined below. Each step is performed by the script or program listed in the step.

1. Convert structure format.

   If the input structure file is in SD format it is converted to Maestro format by `sdconvert`. Parities specified in the SD file are converted into chiralities, which are stored as properties in the Maestro file.

2. Select structures.

   A subset of the input structures can be selected for processing. The selection is done by `maesubset` for Maestro input files and by `sdconvert` for SD input files.

3. Add hydrogen atoms.

   Structures that have implicit hydrogen atoms may need to have hydrogen atoms added before the 3D structures can be minimized. Hydrogen atoms are added in a manner that is consistent with a particular force field. This step is performed by `applyhtreat`, which is the program used by the Hydrogen Treatment panel in Maestro.

4. Remove unwanted molecules.

   If structures have additional molecules included, such as counter ions in salts and water molecules, these may need to be removed. The `desalter` removes all but the molecule containing the most atoms from each structure.

5. Neutralize charged groups.

   Charged groups must be neutralized before ionization states can be generated. Neutral molecules are also required by various applications, such as QikProp. The neutralization is performed by `neutralizer`, which adds or removes hydrogen atoms.

6. Generate ionization states.

   For some applications it is important that all species that exist in a given pH range are available. In this step, the `ionizer` generates various ionization states for each structure. This step should be preceded by a neutralization step.

7. Generate tautomers.

   As with ionization, the significantly populated tautomers may be important for some types of calculations. The `tautomerizer` generates various tautomers for each structure.

8. Filter structures.

   In this step, structures that match specified conditions can be removed. The condition can be on a property, such as Molecular weight > 1000, or on the structure, such as the presence or absence of a specific functional group. This step is performed by `ligparse`.

9. Generate alternative chiralities.

   2D structures do not always have complete chirality information, and it can be useful to vary the chiralities of the atoms to find all the low-energy structures or to provide a range of possible structures for investigation. This step identifies additional chiral atoms in the structures and generates additional structures with the same molecular formula but different chiral properties. The step is performed by the `stereoizer`.

10. Generate low-energy ring conformations.

    When ring conformation information is not available, it is important to generate a range of conformers so that the low-energy structures can be located. Ring confirmations are generated for each structure by `ring_conf`.

11. Remove problematic structures.

    Structures that could cause subsequent processing failures either in the energy minimization of the structures or in other applications are removed by `premin`.

12. Optimize the geometries.

    The geometries of the generated structures are optimized using a restricted version of the MacroModel computational program, `bmin`, or a short conformational search is performed to relax the structure into 3 dimensions while strongly encouraging chiral centers to adopt the proper chiralities (if the structure is highly strained).

13. Convert output file.

    If output in SD format was requested, `sdconvert` is run to perform the conversion.

5.2.2 The LigPrep Panel

The LigPrep panel allows you to set up LigPrep jobs in Maestro. Choose LigPrep from the Applications menu to open the panel. For details of panel options and operation, see Chapter 3 of the *LigPrep User Manual*.

The default options in the LigPrep panel run the `desalter`, add hydrogens, and minimize the ligand structure (performing a 2D-3D conversion, if necessary). Below are notes on panel options that produce more than one output structure per input structure.

The `stereoizer` can generate two stereoisomers per chiral center in the ligand, up to a specified maximum. There are three Stereoisomers options:

The first two options, Retain specified chiralities (the default) and Determine chiralities from 3D structure, generate both isomers only at chiral centers where chirality is unspecified or indeterminate; centers with known chirality retain that chirality.

The difference is that Retain specified chiralities takes its chirality data from the input file (SD or Maestro), while Determine chiralities from 3D structure ignores input file chiralities and takes chirality information from the 3D geometry.

Generate all combinations produces the maximum number of structures, up to the maximum, which by default is 32 stereoisomers, but can be changed using Generate stereoisomers (maximum) *max* per ligand.

The `ionizer` (following the `neutralizer`) can generate all the ligand protonation states that would be found in the specified pH range. The Ionization options are:

Retain original state

Neutralize (best for QikProp)

Generate possible states at target pH *target* +/- *range*. This is the default, and can generate several different output structures for each input structure. The default pH *target* is 7.0 with a +/- *range* of 2.0, so the default pH range is 5.0 – 9.0. Both the target and range settings can be changed.

Generate low energy ring conformations: *number* per ligand. The default is to generate only the lowest energy conformation.

Desalt is selected by default.

Generate tautomers is selected by default. The `tautomerizer` generates up to 8 tautomers per ligand, selecting the most likely tautomers if more than 8 are possible. If you are comfortable that the input structures are already in the correct tautomeric form for docking to a particular target, then the `tautomerizer` should be turned off by deselecting Generate tautomers.

5.3 The Ionization State Expander (`ionizer`)

While LigPrep as a whole requires additional licenses, one LigPrep tool, the `ionizer`, is included with Liaison. This section provides an introduction and usage summary for the `ionizer` as a service.

The `ionizer` generates ionization states of ligands to match the pH range and other conditions that you specify. The resulting ligands can be used as input to programs such as Liaison. Starting with a Maestro-format input file of neutral molecular structures (for example, from a database), the `ionizer` produces a Maestro-format output file that has expanded to include multiple ionization states of each molecule, allowing you to select among them.

The `ionizer` requires the installation of a module called `services`. When you run the `INSTALL` script to install Schrödinger software, be sure to select the `services` product, which contains the `ionizer` software. For more information on installation, see the *Installation Guide*.

The `ionizer` must be run from the command line as follows:

```
$SCHRODINGER/utilities/ionizer [options]
```

The options are listed in Table 5.1.

Table 5.1. Summary of `ionizer` *options*

Option	Description
`-h|-help`	Show usage summary message.
`-doc`	Show more detailed usage message.
`-v|-ver|-version`	Show program version information.
`-j|-job|-jobname` *jobname*	Base name of job. No default (must be specified unless all essential files are specified).
`-i|-in|-infile` *infile*	Default is *jobname*`.mae`.
`-o|-out|-outfile` *outfile*	Default is *jobname*`-ion.mae`.
`-b|-bad|-badfile` *badfile*	Default is *jobname*`-ion-bad.mae`.
`-l|-log|-logfile` *logfile*	Default is *jobname*`.log`; use `-l` to log to screen.

Table 5.1. Summary of `ionizer` *options (Continued)*

Option	Description
`-ph` *value*	Effective pH of active site (default 7.0).
`-pht|-phthresh` *maxdiff*	pH difference threshold (default 2.0). For pH-based ion state rejections, where *maxdiff* is the difference limit on de/protonated |pKa-pH|.
`-pkt|-pkthresh` *maxdiff*	Strong/weak pK threshold (no default). Overrides pH-based rejection mode; reject on pKa values only (no pH), where *maxdiff* is the limit on de/protonated pKa differences.
`-mi|-maxions` *count*	Maximum number of ionizations (default 4).
`-mq|-maxabstotq` *charge*	Maximum absolute total charge (default 2).
`-mg|-maxgroups` *count*	Maximum number of ion groups to handle (default 15).
`-mo|-maxoutcts` *count*	Maximum number of output structures per input structure (default 512).
`-sm|-showmatches`	Show substructure pattern matches.
`-sf|-showfinal`	Show final ionization candidate list.
`-ll|-loglevel` *level*	Expansion report log level: Use `0` for quietest (default). Use `1` to log state generations. Use `2` to log ion fragment fusions too.
`-ss|-showskips`	Show skipped state generations. Augments log level 1 and up. Log levels > 1 give skip reasons.
`-kp|-keep_props`	Retain all properties in output CTs. Absent this option, connectivity-dependent properties are cleared
`-strict`	Terminate run if any input CT is bad. Unsets default fault-tolerant mode. Bad structure file option is ignored.
`-s|-spec|-specfile` *specfile*	Use nonstandard patterns spec file.
`-rw|-retitle_with` *prefix*	Add ion state number onto structure titles.

For a more detailed usage summary, use the command

```
$SCHRODINGER/utilities/ionizer -doc
```

For complete documentation on the `ionizer`, see the `README` file:

```
$SCHRODINGER/services-v<i>version</i>/doc/README.ionizer
```

Running Liaison From Maestro

Liaison is a method of predicting ligand-protein binding free energies using a model that has been fitted to known binding energy values. The process involves two steps, a fitting step and a predicting step. Each step is carried out as two tasks, a simulation task and an analysis task.

The Liaison panel is used to set up and run the simulation tasks. It runs the Ligand & Structure-Based Descriptors script (`lsbd`) to set up and run the Liaison job. The analysis tasks are performed using the Build QSAR Model panel of Strike. For more information on Strike, see the *Strike User Manual*. A tutorial introduction to the Liaison tasks is given in Chapter 3.

Some features of the Liaison panel in earlier versions than 4.0 are not available in the current Liaison panel. For information on using the old Liaison panel, see Appendix A.

6.1 Overview of Liaison Tasks

Before you run a Liaison simulation, you should ensure that the receptor and the ligands are properly prepared, as described in Chapter 4 and Chapter 5. You should also ensure that the ligand structure file includes the known binding energies.

To run a Liaison simulation:

1. Set the job name and select the host and number of procesors in the Job options section of the Systems folder.

   If you want to choose a remote machine or batch queue as the host for the job, ensure that the current working directory is mounted on the remote host. Liaison input files are written to the current working directory. Liaison does not have the ability to copy input files from a local directory to a remote scratch directory.

2. Specify the systems to be simulated in the Specify structures section of the Systems folder. You can take the receptor and ligands from a Glide pose viewer file, or you can take the receptor from the Workspace or a file and take the ligands from the Project Table or a file.

3. Specify the kind of system to be simulated in the Job options section of the Parameters folder.

4. Set constraints if required in the Specify restrained/frozen shells section of the Parameters folder.

5. Click Start.

When the job finishes, the results are incorporated into the Project Table. The scores and the various components that are used in the analysis task are added as properties. If the job does not incorporate, select it in the Monitor panel and click Monitor.

To run a Liaison fitting job:

1. Select the results of a Liaison simulation:
   - If the simulation results are already in the Project Table, select the relevant entries, and choose Build QSAR Model from the Strike submenu of the Applications menu.

     If the simulation results include both the training set and the test set, you should select the training set for the fitting.
   - If the simulation results are not incorporated, click Browse in the Analysis folder of the Liaison panel, navigate to and select the file *jobname*`-final.mae`, and click Fit or Predict in Strike.

   The Build QSAR Model panel of Strike opens, with the Liaison results loaded.
2. Select the descriptors required for the fit from the list (click, shift-click, control-click):
   - For an LRM model, select Uvdw, Uele, and Ucav.
   - For a LiaScore model, select LiaScore.
3. Choose Multiple Linear Regression from the Regression method option menu.
4. For Descriptors, ensure that Use all selected is selected.
5. Click Choose to choose the activity property.

   If the activity property is not in the Project Table, you can import it from a CSV file, but you must do this before clicking Choose.
6. Click Start.
7. Set job options in the Start dialog box, and click Start.

To run a Liaison prediction job:

1. Select the results of a Liaison simulation for the test set (the ligands whose binding energy you want to predict) in the Project Table.
2. Choose Predict from the Strike submenu of the Applications menu.
3. Select the model you want to apply from the list.
4. Click Start.

5. Set job options in the Start dialog box, and click Start.

When the job finishes, you can view the results by clicking View. The predicted values are also added to the Project Table.

6.2 The Liaison Panel

The main part of the Liaison panel consists of three tabbed folders:

- Systems folder
- Parameters folder
- Analysis folder

Below these folders on the left are the Start button, for starting the Liaison simulation, and the Write button, for writing the Liaison input files for later use from the command line.

To open the Liaison panel, choose Liaison from the Applications menu in the main window.

6.2.1 Systems Folder

This folder has two sections, for setting the job options and for defining the system to be studied.

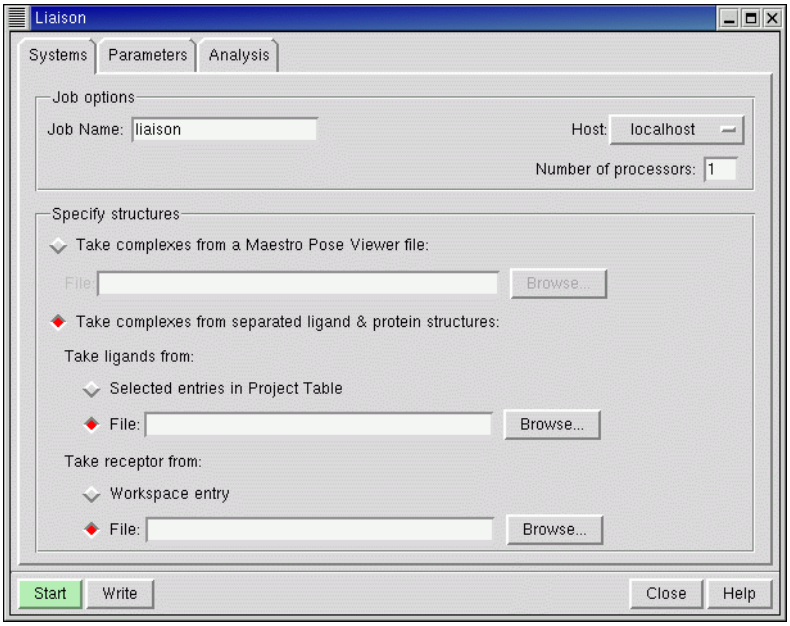


Figure 6.1. The** Systems **folder of the** Liaison **panel.

6.2.1.1 Job options Section

Job Name text box

Specify the name for the job. This name is used to construct file names for the job.

Host option menu

Choose a host on which to run the job. If you choose a remote host or batch queue as the host for the job, ensure that the current working directory is mounted on the remote host. Liaison input files are written to the current working directory. Liaison does not have the ability to copy input files from a local directory to a remote scratch directory.

Number of processors text box

Enter the number of processors you want to use for the job, which will be distributed according to the number of ligands.

6.2.1.2 Specify structures Section

Take complexes from a Maestro pose viewer file option and controls

Use a Maestro (Glide) pose viewer file as the source of the receptor and the ligands. Click Browse to navigate to the file, or enter the path in the File text box.

Take complexes from separated ligand & protein structures option and controls

Specify the receptor and the ligand structures separately.

Take ligands from: Select either Selected entries in Project Table or File. If you select a file, click Browse to navigate to the file, or enter the path in the File text box.

Take receptor from: Select either Workspace entry or File. If you select a file, click Browse to navigate to the file, or enter the path in the File text box.

6.2.2 Parameters Folder

This folder has two sections, one for setting the simulation parameters and one for defining restrained and frozen shells. For details on how the parameters are treated by Impact, see

6.2.2.1 Job options Section

The options that apply to both the complex and the free ligand are in the upper part of this section. Options that apply to one or the other are in folders labeled Ligand Simulation and Complex Simulation in the lower part of this section. The controls in these folders are identical, and are described below.

Sampling method option menu

Choose the method for performing the simulation, from Minimization, Hybrid Monte Carlo, or Molecular Dynamics.

Minimization algorithm option menu

Choose an algorithm for performing the minimization steps in any of the three sampling methods. Available algorithms are Truncated Newton, Conjugate Gradient, and Steepest Descent.

Simulation temperature text box

Specify the temperature of the simulation in K. Default: 300 K Not available with the Minimization sampling method.

Temperature relaxation time text box

Specify the the time scale, in picoseconds, on which heat is exchanged with the heat bath. Default: 10 ps Not available with the Minimization sampling method.

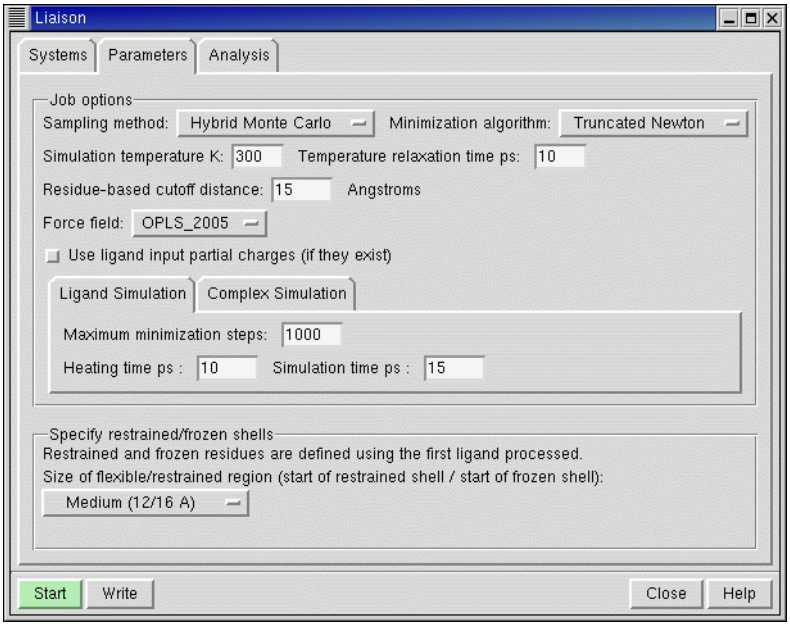


Figure 6.2. The Parameters ***folder of the*** Liaison ***panel.***

Residue-based cutoff distance text box

Set the value (in Å) for the cutoff distance of non-bonded interactions. All pairwise interactions of an atom in one residue with an atom in another residue are included on the non-bonded pair list if any such pair of atoms is separated by this distance or less. Default: 15 Å.

Force field option menu

Select the force field. The force field options are OPLS_2005 (the default) and OPLS_2001. Liaison simulations are run using Surface Generalized Born (SGB) continuum solvation. Select the OPLS_2005 force field if you want the calculation to use the improved parametrized nonpolar model instead of the default SGB terms that are used with OPLS_2001.

Use ligand input partial charges (if they exist) option

Select this option to use the input charges for the ligand in Liaison calculations (for both the free and the bound state).

Maximum minimization steps text box

Specify the maximum steps to take during any minimization. Can be set independently for ligands and complexes. Default: 1000.

Heating time text box

Set the time in picoseconds over which the system is heated before the LIA task is started, in an HMC or MD simulation. Default: 10 ps.

Simulation time text box

Set the simulation time for the LIA task, in an HMC or MD simulation. In this task the averages for the van der Waals, Coulombic, reaction field and cavity terms are determined.

6.2.2.2 Specify restrained/frozen shells Section

Specify the residues that are to be restrained or frozen, by choosing cutoffs for the start of the restrained region and the start of the frozen region. Residues with any atoms inside the restrained region boundary are treated as flexible. Residues with any atoms inside the frozen region boundary but outside the restrained regon boundary are restrained. Residues with all atoms outside the frozen region boundary are frozen. You can choose from a range of options for the location of the two boundaries.

Note: You should make sure that you make the same choice of restrained and frozen shells when you run Liaison for a particular system, otherwise the results will be invalid.

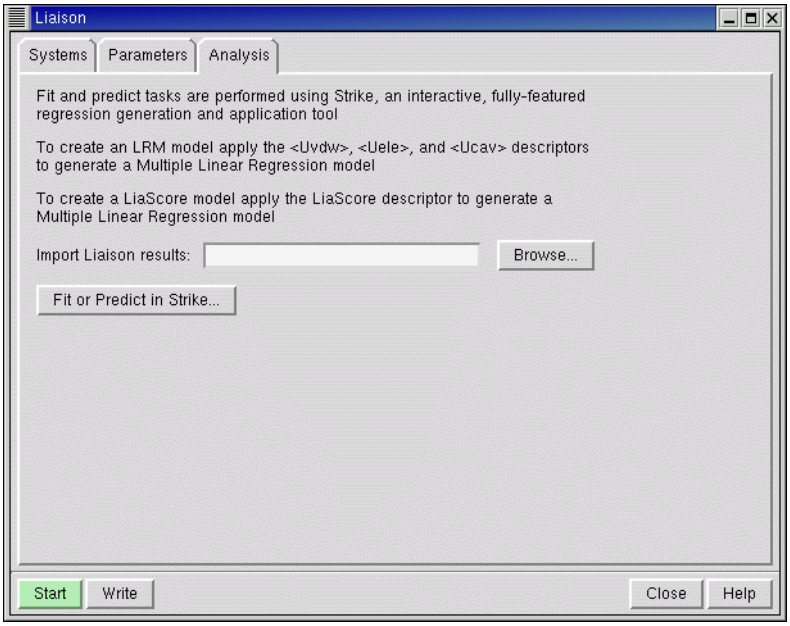


Figure 6.3. The Analysis ***folder of the*** Liaison ***panel.***

6.2.3 Analysis Folder

This folder provides an interface to Strike for performing the fit and predict steps of the Liaison analysis. You can use Strike independently. If you use these controls, you must specify the file that contains the results of the Liaison simulation, named jobname-final.mae. The structures in this file are imported into the Project Table; when you click Fit or Predict in Strike, these entries are selected in the Project Table and used as input to Strike. Clicking this button opens the Strike Build QSAR Model panel. For more information, see Chapter 4 of the *Strike User Manual*.

Command-Line Liaison and Impact

Although you will normally set up jobs using the controls and settings in the Maestro GUI, you can submit Liaison jobs either from within Maestro or from the command line. The same is true for general-purpose Basic Impact calculations and the Protein Preparation facility. Advantages of running from the command line include:

- The command-line scripts can run all full-featured jobs written using the Impact panels in Maestro, and also allow you to override specific run-time values that are not accessible through the Maestro interface.
- Command-line scripts allow you to run jobs when you want.
- Command-line scripts can be modified and jobs can be re-run without reconfiguring and reloading job settings in Maestro.
- Some job options, such as trajectory file analysis, are available only when you run Impact from the command line.

The `SCHRODINGER` environment variable must be set to run jobs. You can define `SCHRODINGER` as follows:

csh/tcsh: `setenv SCHRODINGER` *installation-directory*

bash/ksh: `export SCHRODINGER=`*installation-directory*

Unless otherwise specified, Schrödinger applications and utilities run under a job control system and are automatically run in the background. You need not add an `&` at the end of the commands to have them run and immediately return your command prompt. The `-WAIT` option of the `impact` command forces the shell to wait until the job is finished, so you can embed such commands in other scripts.

7.1 Running Liaison From the Command Line

Liaison uses its own scripts for running simulations from the command line. You can use Maestro to write the scripts. See the Maestro online help or the *Impact Command Reference Manual* for more information.

7.1.1 Liaison Simulation Jobs

You can run Liaison simulations from the command line by using (and editing, if desired) files written by Maestro when you select Write from the lower portion of the Liaison panel. Maestro also creates the necessary directory structure. Alternatively, you can set up the directory structure and job files by hand or with an automated script. If you want to automate the process, use Maestro-written scripts as templates.

To run a Liaison simulation from the command line, change to the Maestro working directory and enter the following command at the shell prompt:

```
./simulate_jobname
```

This command runs the `simulate_`*jobname* script that Maestro writes to the Maestro working directory. The `simulate_`*jobname* script ensures that the "free" and "bound" input files are run for each ligand/receptor pair and are named appropriately. Several relevant lines from such a script are shown below. The script shows that the job covers 5 ligands (Lig1 … Lig5). Note the use of backquotes:

```
JOB_NAME=liaison
JOB_LIST=`echo Lig1 Lig2 Lig3 Lig4 Lig5`
NPROC=1
```

You could run a specific ligand from the command line by altering the `JOB_LIST` line in this script, or in a copy of it. For example, to run only the `1bkm_3m_2` ligand, the `JOB_LIST` line should appear as:

```
JOB_LIST=`echo 1bkm_3m_2`
```

Provided that the requisite directory, structure, and input files exist, additional ligands can be run by adding to the `JOB_LIST` line.

Liaison Simulate jobs, which are more computationally intensive than Analyze jobs, can be run on several processors simultaneously by appropriately setting the number of processors. Enter the desired value in the Number of processors to use text box in the Settings folder of the Liaison panel in Maestro or change the value of the variable `NPROC` in the `simulate_`*jobname* file. For `NPROC` > 1, `NPROC` jobs will run concurrently. When one job finishes, another one will start, until all receptor-ligand pair simulations have been submitted. Of course, you need to have sufficient Liaison licenses to run the requested number of jobs.

7.1.2 Liaison Analysis Jobs

Liaison Fit or Predict Analyze jobs can also be started from the shell prompt by running the `fit_`*jobname* or `predict_`*jobname* script Maestro writes to the Maestro working directory. However, fitting and prediction analysis calculations are virtually instantaneous, once the

prerequisite simulation calculations have finished, and it is usually more convenient to submit such jobs directly from the Maestro interface.

7.2 Running Impact From the Command Line

Basic Impact calculations can be run from the command line using the syntax shown below.

```
$SCHRODINGER/impact [options] [-i] input-file
```

To run `impact`, you must specify the input file, *input-file*. If *input-file* does not end in `.inp`, Impact looks first for *input-file* as specified. If that file doesn't exist, it then looks for *input-file*`.inp`. If `-i` is omitted, then *input-file* **must** end in `.inp` and must be the last argument in the command line. If `-i` is included, the input file specification can be placed anywhere on the command line.

The options that you can specify when initiating jobs from the command line are described in Table 7.1 and Table 7.2. To view the usage summary information, define the `SCHRODINGER` environment variable and enter `$SCHRODINGER/impact -h` in a terminal window.

Table 7.1. `Impact` *command options*

Option	Description
`-h`	Print usage summary and exit
`-v`	Print version number of startup script and exit
`-o` *output-file*	File for writing output and log messages. If this option is omitted, Impact names the log file *jobname*`.log`, where *jobname* is taken from the Impact input file name.
`-s` *size*	Use specific "size" version of the Impact executable. Acceptable values for *size* are `medium` or `huge`. If omitted, `medium` is assumed in most cases; it is valid for up to 8000 atoms or 8000 bonds.
Liaison-Only Options	
`-liasim [-d]` *dir*	Run a Liaison simulation job using ligand directory *dir*. If `-d` is omitted, *dir* must be the last argument on the command line.
`-c` *controlfile*	Specifies name of control file for fit/predict jobs
`-l` *datafile*	Specifies name of data file for fit/predict jobs
`-n` *jobname*	Specifies optional name to use for fit/predict jobs
`-x` *outfile*	Specifies optional output file name for fit/predict jobs

Table 7.2. Schrödinger job control options

Option	Description
`-HOST` *host* `-HOST` *host:n* `-HOST "`*host1 host2*`"`	Specify a remote machine (optionally, its number of processors *n*) on which to run an Impact job. Can also be used to specify a batch queue to submit the job to, or a collection of hosts for distributed or parallel jobs. Default is to run on the local host. See Section 7.1.
`-USER` *user*	Specify a remote user name to run Impact job under. Default is to use the local user name.
`-WAIT`	Do not return the command prompt until job finishes. This is useful in command scripts in which you have specified actions to take only after the Impact job finishes.
`-WHICH`	This switch is a diagnostic tool printing the available Impact installations you can use for the local machine. The job itself is not submitted. The first one listed is the default path; the options `-REL`, `-VER`, and `-ARCH` can direct your job to use a different installation.
`-REL` *release*	This option selects a specific version number of Impact to use. The default is the latest (highest number). Formats like `-REL v4.0`, `-REL 35000`, and `-REL 27` are supported.
`-VER` *pattern*	If you have multiple installations installed, you can specify a pattern with the `-VER` option that matches the installation path to use for your job. The default installation is the one printed by `-WHICH`.
`-ARCH` *platform*	If you have more than one architecture installed for a given system, e.g., AIX-com and AIX-pwr3, then this flag can be used to select either of them, such as `-ARCH pwr3`.
`-LOCAL`	Force remote jobs to run in a local directory, rather than on the remote host. Only active when `-HOST` is used.

Note: The default molecular mechanics force field for Basic Impact applications is the OPLS2001 version of OPLS-AA. OPLS2003 is also available. OPLS2001 and OPLS2003 are designed to work with automatic atom-typing, and are incompatible with template mode. If you attempt to write a template file (Impact command `WRITE TEMPLATE`) while using OPLS2001, an error message is displayed to remind you that this command can only be used with OPLS1999 or OPLS2000 force fields. To use one of these older force fields, add a line to your input file before the `CREATE` task, for example:

```
SET FFIELD OPLS1999
```

For distributed processing, the `run_jobs.pl` script can be used. (Distributed processing for Basic Impact calculations is not available from the GUI.)

7.3 File Name Conventions

7.3.1 Location of Files and Working Directory

For Basic Impact applications, Maestro normally writes input files to the directory from which you launched Maestro (the *Maestro working directory*). Impact also normally writes its output files to the same location. Liaison is an exception: a directory hierarchy is created based on job names and on the names you assign to the individual ligands being simulated (see Figure A.6 on page 119 and Section A.4.1 on page 105).

7.3.2 Liaison and Impact File Names

A typical Impact job has one command-script file (*jobname*`.inp`), one or more structure files (*jobname*`.mae`, *jobname*`.pdb`, or *jobname*`.sdf`), and after execution, several output files (e.g., *jobname*`_out.mae` for structure files and *jobname*`.out` for textual data).

If a file already has the name of an output file, in many cases Impact renames the old file with a numerical extension (*filename*`.out.01`, *filename*`.out.02`, and so on) for archival purposes. The new job's output is then written to the base name (*filename*`.out`). If you do not need the old files, you can remove them.

Some files, such as *jobname*`.log` files, are newly written each time Impact runs a calculation. Likewise, old *jobname*`_pv.mae` files are overwritten. Other examples of files that are *not* incremented are:

- *jobname*`_out.mae` structure files, for Basic Impact minimization jobs.
- *jobname*`_lig_min.mae` and *jobname*`_rec_min.mae` files, for the minimization section in Liaison.
- *jobname*`_rec_fin.mae` and *jobname*`_lig_fin.mae` files, for the dynamics and HMC sampling methods in Liaison.

In addition, *jobname*`_out.mae` files are *not* produced by default for Liaison jobs—*jobname*`_min.mae` and/or *jobname*`_fin.mae` files are written instead.

Table 7.3 contains descriptions of the various file types. For more information, see the Maestro online help or the *Impact Command Reference Manual*.

Table 7.3. Liaison file extensions

Extension	Description
`.inp`	Impact input file or script. Impact input files are formatted plain-text files written in the Impact input file language, DICE. Maestro creates Impact input files before job submission, or you can create or edit them manually with a text editor.
`.mae`	A Maestro format structure file, a plain-text file written by Maestro containing atom, bond, and other information for one or more molecules.
`.log`	An Impact log file. If specified, a `.log` file captures standard output and standard error messages in text form. This file is overwritten during subsequent runs.
`.out`	An Impact output file containing information similar to that found in log files (no standard error). Output files are appended with numerical extensions when the input file is run again. Up to 99 output files are retained.
`.01`, `.02`, etc.	A file containing results from previous Impact calculations run from the corresponding *jobname*`.inp` file.
`_out.mae`	An Impact output structure file written in the Maestro file format. Liaison and some Impact jobs do not write *`_out.mae` output structure files.

7.4 Job Control for Impact and Liaison

Once your jobs are launched, you can monitor their progress using the Monitor panel in Maestro. The command `$SCHRODINGER/jobcontrol` can also be used. It has many options, which are summarized below. The two most useful options are:

```
$SCHRODINGER/jobcontrol -list
```

which will show the status of all your jobs, and:

```
$SCHRODINGER/jobcontrol -kill
```

to terminate any job and its subjobs, if any exist. The command format is:

```
$SCHRODINGER/jobcontrol action [job_selection]
```

where *action* is one of the following and *job_selection* specifies one or more jobs. The action will be applied to each selected job.

`-list`	List the JobId, job name and status. By default, lists all active jobs.
`-show`	Show basic information about the job
`-kill`	Terminate the job immediately
`-stop`	Terminate the job as soon as possible
`-pause`	Suspend the job temporarily
`-resume`	Continue running a paused job
`-monitor` *n*	Ask for monitoring files to be sent every *n* sec
`-cancel`	Cancel a job that has been launched, but not started
`-purge`	Remove completed job from the database
`-help`	Produce usage summary with query construction examples

The *job_selection* argument consists of one or more JobIds, job names, status codes, or queries. This field is optional; if *job_selection* is omitted, the default selection is the query `status!=completed`, that is, all active jobs. It can also be the word `all`, to select all jobs in the jobs database.

For more information, see the *Job Control Guide*. For an introduction to running and monitoring jobs in Maestro, see Section 2.10 on page 28.

The Job Control facility may be used to manage and, if necessary, kill Liaison jobs. This facility can be invoked from the Maestro Monitor panel. To kill a Liaison job, select the *jobname*`_sim` entry and click the Kill button, and all the subjobs will quit as well.

For the purpose of killing Liaison jobs, the `-list` or `-show` actions can be used to list the jobs in the jobs database, and `-kill` *jobname* or `-kill` *jobid* can then be used to kill one of these jobs. The top-level simulation script is called `simulate_`*jobname*. This script launches a job named *jobname*`_sim`. Use the *JobId* corresponding to this job as the argument to `jobcontrol -kill` *jobid*—all the individual ligand sub-jobs are killed as well. The job name *jobname*`_sim` can be substituted for the JobId in the `kill` command.

For more information on the Job Control facility, see the *Job Control Guide*.

7.5 Command-Line Applications and Utilities

Several Liaison support modules are command-line applications or utilities.

Utilities are located in the directory `$SCHRODINGER/utilities`. You may want to add this directory to your path so that they are easy to run by name from the command line. For usage summary information on command-line applications and utilities, use the `-h` (help) option:

```
$SCHRODINGER/protprep -h
$SCHRODINGER/utilities/utilityname -h
```

Protein preparation jobs can be run from the command line using the `protprep` application:

```
$SCHRODINGER/protprep [options] input-file
```

The command-line application `protprep` is located in the main Schrödinger directory. You can also use the `pprep` and `impref` utilities. See Section 4.11 on page 60 for more information about command-line protein preparation.

Technical Notes for Liaison

8.1 Models of Ligand Binding

The typical binding site for a ligand is the active-site cavity of a protein receptor. When no ligand is present, this cavity is filled with water molecules. When a ligand binds to the protein in this cavity, it displaces water molecules in the active site, which return to bulk solvent.

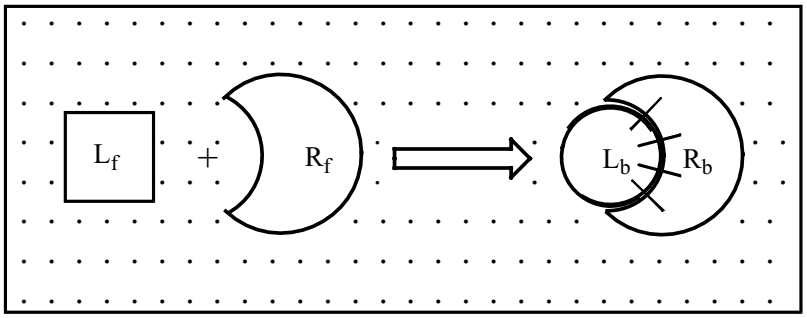


Figure 8.1. Schematic view of ligand binding in a receptor cavity with displacement of water. The ligand and the receptor both change, to a greater or lesser extent, from a "free" (f) to a "bound" (b) conformation.

The principal factors determining the strength and specificity of binding are as follows:

- The degree to which hydrophobic groups on the ligand interact with hydrophobic pockets or patches on the protein surface to release water into the bulk. This release is favorable both energetically (more hydrogen bonds are formed by the released water molecules) and entropically (the released waters are less constrained orientationally and are no longer confined to a restricted cavity).
- The extent to which the ligand forms hydrogen bonds or metal ligations in hydrophilic regions with appropriately placed polar or charged groups on the receptor. Such complementarity is essential for achieving adequate binding affinity and specificity.
- The ease with which the ligand fits into the protein cavity. An important question is what it costs the ligand (and the protein) in energy and/or entropy to accomplish this fit—i.e., to change from the free to the bound conformation, as indicated schematically in Figure 8.1. Energy will be required if the ligand has to be distorted away from its naturally preferred, low-energy conformation into a higher-energy conformation when it

binds to the receptor. At the same time, entropy will be lost if the ligand is very flexible in solution and then is confined to a small number of conformations in the receptor cavity. Both effects, as well as similar restrictions on the conformation of protein side chains, act against binding.

8.1.1 Limitations of Free-Energy Perturbation

Molecular simulation methods have been used to calculate binding free energies of protein-ligand calculations since the pioneering applications of free energy perturbation (FEP) approaches by McCammon, Kollman, Jorgensen, and others approximately 20 years ago. During the past two decades, many FEP calculations have been carried out in academic groups and in pharmaceutical and biotechnology companies. But while notable successes have been achieved, FEP methods are in limited use in drug-discovery projects, for several reasons:

- FEP calculations are typically limited to small changes in ligand structure, restricting the applicability to the very last phase of lead optimization.
- FEP calculations are very expensive computationally, and often cannot be completed on a time scale compatible with the schedule of a given drug-discovery project.
- Inaccuracies in force fields and sampling methods can lead to errors in FEP predictions.

8.1.2 Advantages of Linear Response Methods

The limitations of FEP motivated the development of linear-response (LR) methods by Aqvist several years ago (Hansson, T.; Aqvist, *J. Protein Eng.* **1995,** *8,* 1137–1145). Since that time, studies by Jorgensen and others have shown that LR methods can effectively address the above difficulties. In comparison with the FEP approach, the advantages of LRM are as follows:

- In contrast to FEP, where a large number of intermediate "windows" must be evaluated, LRM requires simulations only of the ligand in solution and the ligand bound to the protein. The idea is that one views the binding event as replacement of the aqueous environment of the ligand with a mixed aqueous/protein environment.
- Only interactions between the ligand and either the protein or the aqueous environment enter into the quantities that are accumulated during the simulation. The protein-protein and protein-water interaction are part of the "reference" Hamiltonian, and hence are used to generate conformations in the simulation, but are not used as descriptors in the resultant model for the binding free energy. This eliminates a considerable amount of noise in the calculations—for example, that arising from variations in the total energy that result because slightly different geometries of the protein are obtained for each ligand molecule simulated.
- As long as the binding modes of the ligand are fundamentally similar, LRM calculations

can be applied to ligands that differ significantly in chemical structure.

- LRM calculations are less computationally expensive than FEP calculations.
- The LRM approach allows the binding-energy model to be calibrated by using a training set of compounds for which experimental binding affinities are known. The use of the energy terms as descriptors in the fitting equation introduces an empirical element that allows some of the limitations in the theoretical framework (for example, the neglect of the cost in energy and entropy of fitting the ligand into the protein site) and the physical representation (as reflected by errors in the force field or solvation model) to be partially absorbed into the parameterization. Moreover, some of the steps involved in the binding event, such as the removal of water from the protein cavity and subsequent introduction of the ligand, are not inherently linear. If the linear-response approximation was rigorously valid, the coefficients of the terms would each be 0.5, corresponding to the mean-value approximation to the "charging" integral. In practice, optimization of the fitting parameters yields coefficients that are significantly different from the ideal value of 0.5. This empirical element sacrifices generality: the method probably requires the ligands to have similar binding modes, and new parameters must be developed for each receptor. In return, one can obtain a reasonable level of accuracy with a modest expenditure of CPU time, under assumptions that are quite reasonable for many structure-based drug-design projects.

8.1.3 Advantages of Liaison

Liaison—Schrödinger's continuum-solvent implementation of LRM—has a number of highly attractive features in addition to those listed in Section 8.1.2:

- The use of the Surface Generalized Born (SGB) continuum model greatly speeds the calculation because the various interaction terms converge much faster than in an explicit-solvent simulation. As a result, the required CPU time is reduced by a factor of 10 or more.
- An even greater reduction in computational effort can be achieved by using a simple energy minimization protocol, rather than a molecular dynamics or Hybrid Monte Carlo simulation, and obtaining the LRM fitting data from the lowest-energy point reached by the minimization. While there is sometimes a small degradation of accuracy as compared to a simulation, the speed of the calculation is qualitatively enhanced.
- When "energy-minimization" sampling is coupled with a highly efficient Truncated Newton minimizer, Liaison calculations are fast enough to be applied routinely in a contemporary drug-discovery context. This approach makes Liaison very attractive for screening a large number of ligands.

- Schrödinger's automatic atom-typing scheme for the OPLS-AA force field (Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. *J. Am. Chem. Soc.* **1996,** *118,* 11223–11235) assigns charges, van der Waals, and valence parameters with no human intervention. A key feature of OPLS-AA is that, via fitting to liquid-state simulations, excellent reproduction of condensed-phase properties is obtained.

The Maestro graphical user interface makes it easy to set up and run Liaison calculations. First, a training set of compounds for which experimental binding affinities are available can be used to generate simulation data that are then employed to determine the optimal LRM parameters. Once the parameters have been determined, libraries of compounds with unknown binding affinities can be run and their binding affinities can be predicted. Alternatively, Liaison can perform scoring on relaxed protein-ligand complexes, allowing direct prediction of ligand binding energies.

It is important that the protein structures be correctly prepared. See Chapter 4 for a description of Schrödinger's protein-preparation facility.

8.2 LiaisonScore Binding Free Energy Model

Liaison calculates a scoring function, similar to GlideScore 3.5 SP, over the course of the LRM simulation. This function, previously known as "GlideScore in Liaison" is now called LiaisonScore. The average LiaisonScore found in the simulation can be used to predict binding energies using the formula

$$\Delta G = a(\langle LiaisonScore \rangle) + b$$

instead of the LRM equation described above. The LiaisonScore binding energy model can be selected in the Analysis folder. The Analyze task can then be used to derive values for the a and b fitting coefficients.

This capability allows Liaison to be employed in the early stages of a drug-discovery project, i.e., before a set of ligands with known binding affinities suitable for training a LRM model is available. This approach transcends scoring in Glide itself because it allows the protein site to relax and enables the ligand to undergo full (not just torsional and rigid-body) optimization.

Advantages of LiaisonScore

Recall that docking with Glide is normally done with the vdW radii of non-polar ligand and/or protein atoms scaled back by 10 to 20%. This rescaling creates more room in the rigid protein site and implicitly allows for "breathing motions" a protein site may carry out to accommodate a ligand that is slightly larger in some dimension than the ligand co-crystallized with the protein. But while some protein sites may readily expand (or contract) upon ligand binding, others may be less adaptable, and the device of rescaling radii used by Glide will miss this

distinction. Moreover, Glide further compensates for the limitations of docking into a rigid protein site by allowing intramolecular contacts within the docked ligand pose to be shorter than are physically realistic. These two aspects mean that Glide at times will give good scores to ligands that are too large to bind in a given receptor site. There can also be cases in which the rigid site is somewhat too large for an active binder, particularly with the scaling down of non-polar radii. As a result an active ligand may score more poorly than it should.

In principle, such false positives and false negatives can be eliminated by repeating the Glide scoring after relaxing the ligand-protein complex. In this way, if the protein can readily adapt to the true (not Glide compacted) size and shape of the docked ligand, the recomputed GlideScore value should be good. But if the protein site cannot respond appropriately, a poor re-scored GlideScore value should be obtained. Note, however, that sites that are much too small will simply generate incorrect docked poses, and minimization will not transform such a pose into a correctly docked structure.

Calibration of GlideScores

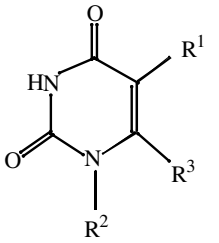
If a training set of ligands with known binding affinities is available, calibrated GlideScore values can be obtained as

$$\text{GlideScore(calibrated)}_i = a\ \text{GlideScore(raw)}_i + b \quad (1)$$

Calibration has no effect on the rank order of calculated binding affinities or on the computed correlation coefficient, but can place the computed GlideScores on the correct experimental scale and make it easier to interpret the calculated values.

8.3 Application to HIV Reverse Transcriptase

The Jorgensen group has been a leader in the development and testing of linear-response approaches for systems of pharmaceutical interest using explicit-solvent methods. An initial application to HIV reverse transcriptase examined the binding of 20 HEPT and 20 nevirapine analogs (*J. Med. Chem.* **2001,** *44,* 145–154). More recently, this work has been extended to encompass 200 different inhibitors covering 8 chemical classes (*J. Med. Chem.* **2002,** *45,* 2970–2987). Nevirapine is a FDA-approved anti-HIV drug of the non-nucleoside inhibitor class, and one of the HEPT analogs (MKC-442) has been in clinical trials. As shown in Figure 8.2, the variation in the R^2 sidechain is particularly sizable, ranging from hydrogen to methyl benzyl ether. It would be very difficult, and very time consuming, to examine structural variations of this magnitude using free-energy perturbation methods.



R^1 = Me, Et, i-Pr, c-Pr

R^3 = SPh, CH_2Ph, OPh, SPH-3,5-di-Me

R^2 = H, Me, Et, Pr, CH_2OCH_3, $CH_2OCH_2CH_3$, $CH_2OCH_2CH_2OH$, CH_2OCH_2Ph

Figure 8.2. Substituents for the HEPT series of inhibitors for HIV-RT.

The Liaison calculations presented in this section use input geometries for the ligands obtained from Glide 3.5 SP and XP dockings using Glide 3.5. Energy-minimization sampling was used in each case. Calibrated GlideScore values using Glide SP and Glide XP, LiaisonScores, and Liaison LRM-model predictions appear in the figures that follow.

Figure 8.3 shows the result obtained in the Glide 3.5 SP dockings, while Figure 8.4 shows the results obtained using Glide 3.5 XP scoring. As can be seen, SP Glide gives only a very rough correlation of the docked GlideScore values (shown as the calculated ΔG_{bind} values) with the experimental binding affinities. This is not surprising, as correlating experimental binding affinities will be beyond the capabilities of Glide docking and scoring in some cases, because of the limitations of rigid-receptor docking discussed above. Figure 8.4 shows that XP docking and scoring gives a considerably better correlation.

The results of applying the LiaisonScore binding energy model, using energy-minimization sampling and starting from the Glide 3.5 SP docked geometries for the ligands, are shown in Figure 8.5. The correlation is much tighter and the largest outliers are gone, showing that relaxation of the protein and docked ligand geometries has improved the results. These GlideScore values, unlike those in the previous two figures, have been put on the scale of the experimental binding affinities by applying the linear transformation shown in Equation (1). Thus, the predicted values now cover the approximately same range as the measured values. The slope of the fitted least-squares line is still less than 1, but is larger than those seen in Figure 8.3 and Figure 8.4.

The conventional LRM fit, again using energy-minimization sampling and starting from the Glide 3.5 SP ligand poses, is shown in Figure 8.6. As can be seen, LRM scoring produces a higher r^2 correlation coefficient and a lower RMS deviation between the calculated and observed binding affinities.

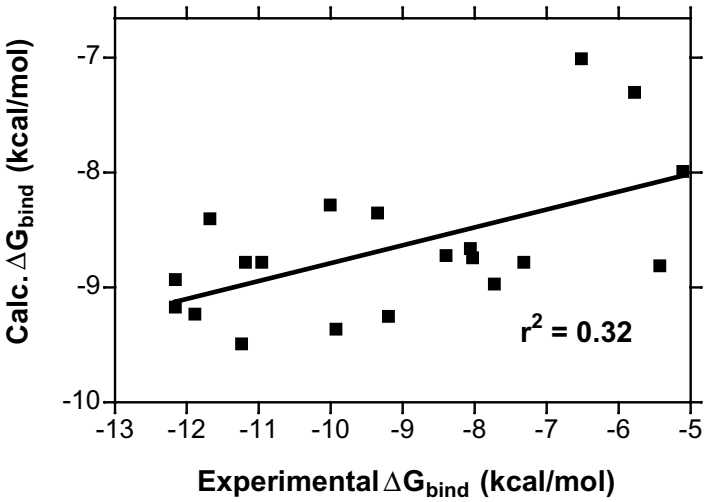


Figure 8.3. Correlation of calculated (GlideScore) and observed binding affinities for Glide 3.5 SP docking of HEPT inhibitors into HIV reverse transcriptase.

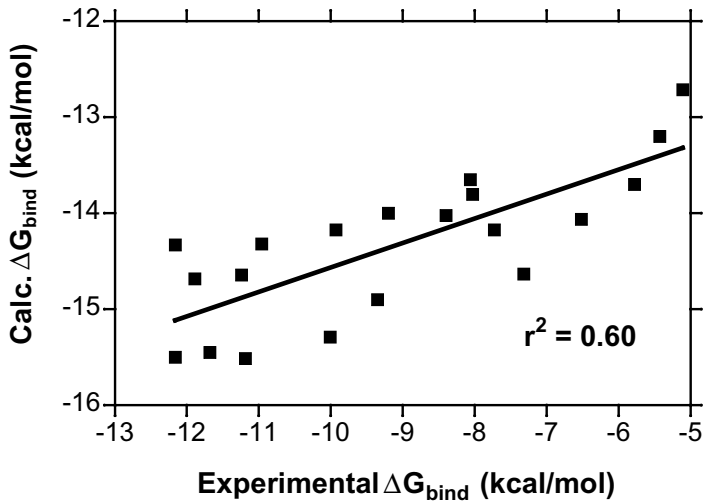


Figure 8.4. Correlation of calculated (GlideScore) and observed binding affinities for Glide 3.5 XP docking of HEPT inhibitors into HIV reverse transcriptase.

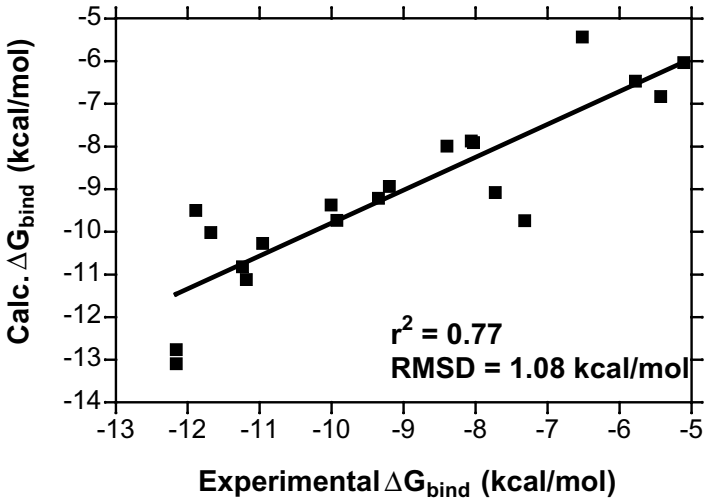


Figure 8.5. Correlation of LiaisonScores for the HEPT inhibitors for HIV reverse transcriptase, calculated by energy-minimization sampling starting from Glide 3.5 SP docked geometries.

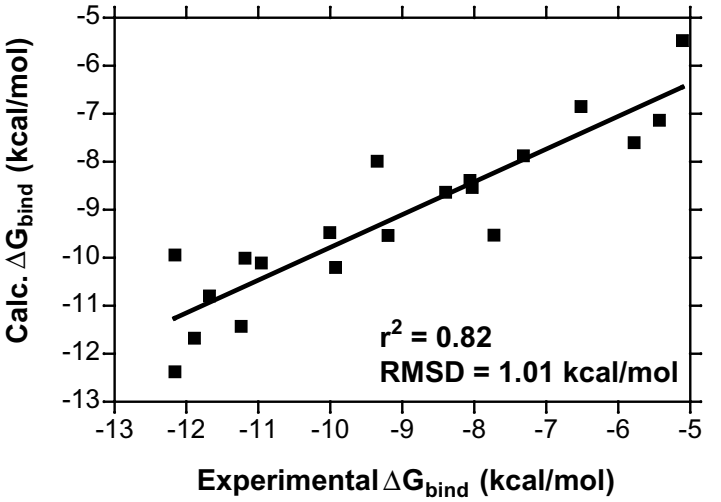


Figure 8.6. Correlation of Liaison LRM scoring for the HEPT series of inhibitors for HIV reverse transcriptase, calculated by energy-minimization sampling starting from Glide 3.5 SP docked geometries.

These results suggest that some of the limitations expected for rigid-receptor docking can be addressed by relaxing the docked complexes before scoring them with GlideScore or via a standard LRM fitting model. LRM scoring gives the best results in this instance, but it remains to be seen whether this will always be the case.

Normally we would expect even better results from energy-minimization sampling followed by rescoring in Liaison when starting from the XP-docked poses. This indeed is our recommended procedure. In this instance, however, the results obtained using the same two models gives correlations that are better than those shown in Figure 8.3 but are not as good as those in Figure 8.4 and Figure 8.5. This is counterintuitive, especially in view of the fact that XP docks only one ligand in a manner that is inconsistent with the other cases, and here the variation merely consists of a reversal of the positions adopted by the pseudo-symmetric C-R1 and N-R2groups (see Figure 8.2). In contrast, SP Glide docks three ligands in an inconsistent, and presumably incorrect, manner.

8.4 Application to ß-Secretase (BACE) Inhibitors

Tounge and Reynolds (*J. Med. Chem.* **2003,** *46,* 2074–2082) recently reported the application of Liaison to ten β-secretase inhibitors of the general structure shown in Figure 8.7, where R_1 is typically a BOC-capped dipeptide, and R_2 and R_3 are usually methyl or isopropyl. These neutral inhibitors have relatively high molecular weights and resemble HIV protease ligands in structure. Tounge and Reynolds also studied two nanomolar co-crystallized octapeptides that have two or three aspartate or glutamate residues and hence are negatively charged at physiological pH.

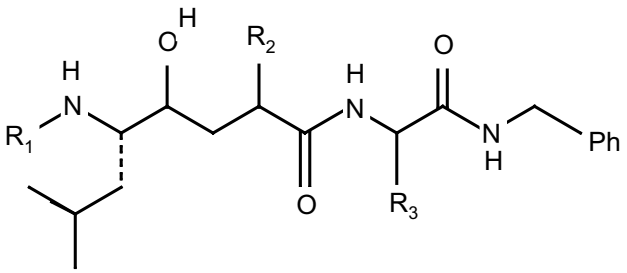


Figure 8.7. Schematic structure of neutral BACE inhibitors studied by Tounge and Reynolds.

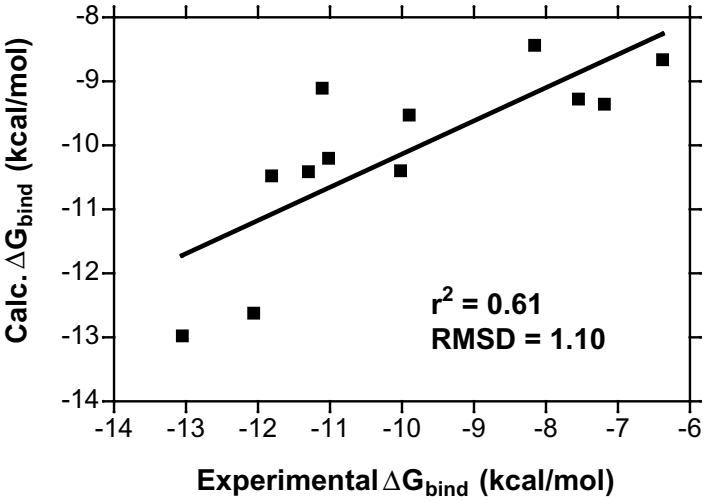


Figure 8.8. Correlation of predicted and observed binding affinities for BACE inhibitors using LRM scoring and energy-minimization sampling.

A plot of their data for energy-minimization sampling, taken from Figure 5 in their paper, is presented in Figure 8.8; hybrid Monte Carlo sampling gave a similar result. The correlation is reasonably good, but the negatively charged octapeptides (the two data points with the most negative calculated and observed binding affinities) fall off the line. This is not surprising, as it is notoriously difficult in binding energy correlations to get ligands having different charge states to fall on a common line.

Tounge and Reynolds also presented results for other simulation conditions that gave poorer fits to the experimental data. We nevertheless regard their results as promising, and intend to employ their series of BACE inhibitors in future efforts to improve Liaison.

Getting Help

Schrödinger software is distributed with documentation in PDF format. If the documentation is not installed in `$SCHRODINGER/docs` on a computer that you have access to, you should install it or ask your system administrator to install it.

For help installing and setting up licenses for Schrödinger software and installing documentation, see the *Installation Guide*. For information on running jobs, see the *Job Control Guide*.

Maestro has automatic, context-sensitive help (Auto-Help and Balloon Help, or tooltips), and an online help system. To get help, follow the steps below.

- Check the Auto-Help text box, which is located at the foot of the main window. If help is available for the task you are performing, it is automatically displayed there. Auto-Help contains a single line of information. For more detailed information, use the online help.
- If you want information about a GUI element, such as a button or option, there may be Balloon Help for the item. Pause the cursor over the element. If the Balloon Help does not appear, check that Show Balloon Help is selected in the Help menu of the main window. If there is Balloon Help for the element, it appears within a few seconds.
- For information about a panel or the folder that is displayed in a panel, click the Help button in the panel. The Help panel is opened and a relevant help topic is displayed.
- For other information in the online help, open the Help panel and locate the topic by searching or by category. You can open the Help panel by choosing Help from the Help menu on the main menu bar or by pressing CTRL+H.

  To view a list of all available Liaison–related help topics, choose Liaison from the Categories menu of the Categories tab. Double-click a topic title to view the topic.

If you do not find the information you need in the Maestro help system, check the following sources:

- *Maestro User Manual*, for detailed information on using Maestro
- *Maestro Command Reference Manual*, for information on Maestro commands
- *Impact User Manual*, for detailed information on basic Impact tasks and panels
- *Impact Command Reference Manual*, for Impact command syntax
- *Strike User Manual*, for detailed information on using Strike
- *Ligand and Structure-Based Descriptors*, for information on using Liaison for descriptors
- Frequently Asked Questions pages, at https://www.schrodinger.com/Liaison_FAQ.html

The manuals are also available in PDF format from the the Schrödinger Support Center. Information on additions and corrections to the manuals is available from this web page.

If you have questions that are not answered from any of the above sources, contact Schrödinger using the information below.

E-mail: help@schrodinger.com
USPS: 101 SW Main Street, Suite 1300, Portland, OR 97204
Phone: (503) 299-1150
Fax: (503) 299-4532
WWW: http://www.schrodinger.com
FTP: ftp://ftp.schrodinger.com

Generally, e-mail correspondence is best because you can send machine output, if necessary. When sending e-mail messages, please include the following information, most of which can be obtained by entering `$SCHRODINGER/machid` at a command prompt:

- All relevant user input and machine output
- Liaison purchaser (company, research institution, or individual)
- Primary Liaison user
- Computer platform type
- Operating system with version number
- Liaison version number
- Maestro version number
- mmshare version number

Running Liaison with the Old Liaison Panel

While the Liaison panel in Liaison 4.0 streamlines the process of running Liaison calculations, some of the flexibility in setting up simulations is not available. As a service to users who want this flexibility, the old Liaison panel has been retained, and can be opened by entering the following command in the command input area in the main window:

```
showpanel liaison
```

In this old panel, the process of running Liaison involves two steps, a fitting step and a predicting step. Each step is carried out as two tasks, a simulation task and an analysis task, as shown in this outline:

1. Fit a binding energy model to a set of training ligands:
    a. Simulate binding for the training set.
    b. Analyze the simulation results to generate the fitted model.
2. Predict binding energies for new ligands:
    a. Simulate binding for the new ligands.
    b. Analyze the simulation results using the fitted model to predict binding free energies for the new ligands.

The use of the old Liaison panel in Maestro to carry out this workflow is described in this appendix.

A.1 The Liaison Panel

The Liaison panel is used to set up and run the Simulate and Analyze results of earlier simulations tasks for the Fit step and for the Predict step. To open the Liaison panel, choose Liaison from the Applications menu on the Maestro main menu bar.

In the upper part of the Liaison panel are tabs for five folders:

- Settings
- System
- Parameters
- Constraints
- Analysis

These folders are discussed in detail in later sections.

In the lower part of the Liaison panel, the following buttons, common to most Maestro panels, appear:

Start

Click the Start button to open the Start dialog box. Section A.2 summarizes starting jobs using the Start dialog box.

Write

The Write button writes out all the files required for the job; however, the job will not actually be started. Once the run files (an input file, *jobname*`.inp`, and one or more input structure files, *jobname_structure*`.mae`) are written by Maestro, the job can be run from the command line using the syntax:

```
$SCHRODINGER/impact -i jobname.inp
```

where *jobname*`.inp` is the input file for the job in question. The log output will be written to *jobname*`.log` by default; a different filename can be specified via `-o` *othername*`.log`.

Type `$SCHRODINGER/impact -h` for a usage summary of the `impact` command, or see Chapter 7 for a discussion of running Impact from the command line.

Close

The Close button, which is located on all Maestro panels, dismisses the Liaison panel without starting the job or writing any files.

Help

The Help button, also on all panels, opens the Help panel with the Liaison panel help topic displayed.

A.2 Setting Up and Starting Jobs

A typical Liaison workflow, in which a binding energy model is fitted to data and then used to make predictions, involves running multiple jobs. This section describes running Liaison jobs in general. Details of running particular steps and tasks are described in later sections.

To set up a Liaison job, start by setting the job type in the Settings folder. The default job type is Simulate. To set up a new simulation job, use the System and Parameters folders; optionally, set atom constraints (frozen atoms or constrained atoms) in the Constraints folder. To analyze previously calculated simulation results, select job type Analyze results of earlier simulations.

The Analysis folder becomes available, and the System, Parameters, and Constraints folders become unavailable.

Some of the options in the Liaison panel require external data. You may need to specify data files that you have created, or identify the location of structure files or previously generated simulation data. These files and other external data are described with the options used to specify them in Section A.4 and Section A.7.

A.2.1 The Liaison Start Dialog Box

When you have specified the required settings and files, and are ready to start the Liaison job, click the Start button in the lower left corner of the Liaison panel. This opens the Liaison Start dialog box. The same dialog box appears whether the step you are in is Fit or Predict, and whether the job type is Simulate or Analyze. It has the following job options:

Name

Type the name of the job, *jobname*, in this text box, or accept the default name. When a job is started or when job files are written using the Write button, Liaison creates a master run directory called *jobname*, if it does not already exist, in the Maestro working directory. This is used as the base directory for ligand directories and other files associated with the job. The default name for both Simulate and Analyze jobs is `liaison`.

The Analyze task that uses the results of a Simulate task needs to use the same master run directory as the Simulate task, so it must be given the same name. For example, if the name of the simulation job was `fit_sim`, the master run directory will be *workingdir*`/fit_sim`, so the name of the analysis job must also be `fit_sim`.

However, Fit jobs and Predict jobs should have different job names. When you have completed the Simulate and Analyze tasks for the Fit step of the Liaison process, use a different job name for the Predict step Simulate task (e.g. `predict_sim`), then use `predict_sim` for the prediction Analyze task as well.

The job name should not be the same as the name of any file that is the source of ligand structures for a simulation job.

Host

Choose a host if you want to run the job on a remote machine. This option menu displays all the hosts defined in the `$SCHRODINGER/schrodinger.hosts` file, with the number of processors on the host in parentheses. The default is `localhost`.

Note: If you run a Liaison job on a remote host, you need to ensure that the job files and directories, which by default are created within the Maestro working directory, are accessible from the remote machine.

Username

Enter your user name, if it is required for running the job on remote machines. The default value is the user name of the user who started Maestro. If this user name is not correct for the selected host, you can change it in this text box. If the job is running locally, this text box is ignored.

Once you have finished setting the options in the dialog box, you can click the Start button to start the job.

A.2.2 Running Liaison Jobs

When you click Start in the Liaison Start dialog box for a Simulate job, Maestro writes and dispatches a script file called `simulate_`*jobname*. If the job type is Analyze results of earlier simulations, the script file is named `analyze_`*jobname*. If you have chosen to Write job files instead, the script file can be used to run Liaison from the command line.

When the job is launched, the Monitor panel appears. For Liaison multiple-ligand simulation jobs, a subjob, *ligandname*, is created for each ligand. Select the subjob in the Monitor panel to view the log file *ligandname*`.log`. (Separate `bound` and `free` log files are written to disk.)

For more information about Liaison directories, see Section A.8.

It is sometimes necessary to use the Schrödinger `jobcontrol` utility, rather than the Monitor panel alone, to manage Liaison jobs and subjobs. For example, if you want to kill a suite of simulation jobs, selecting the parent job in the Monitor panel and then clicking Kill may not kill all subjobs. The `jobcontrol` command-line utility can be used instead, first to list all the subjobs, then to kill them by jobID. The syntax is

```
$SCHRODINGER/jobcontrol action [jobIDselection]
```

Use the `-help` option to see all the possible actions.

To get a listing of all the active jobs and subjobs you have, enter:

```
$SCHRODINGER/jobcontrol -list
```

The top-level job is called *jobname*. Find the jobID corresponding to this job, then enter:

`$SCHRODINGER/jobcontrol -kill` *jobID*

to kill the top-level job and all subjobs underneath it.

A.3 The Settings Folder

The Job type option menu at the top of this folder selects Simulate or Analyze as the Liaison task. This folder also specifies the number of processors to use (subject to the number of Liaison licenses) in concurrent Liaison simulations on a multi-processor machine.

The selection options in the Settings folder are:

- Job type (buttons)
- Number of processors to use (text box)

Job type

This pair of buttons determines the type of Liaison job to be run.

- Simulate

  This is the default job type. After you have run one or more simulations, you can set up and run analysis jobs. Simulations are set up using the System and Parameters folders, and, optionally, the Constraints folder.

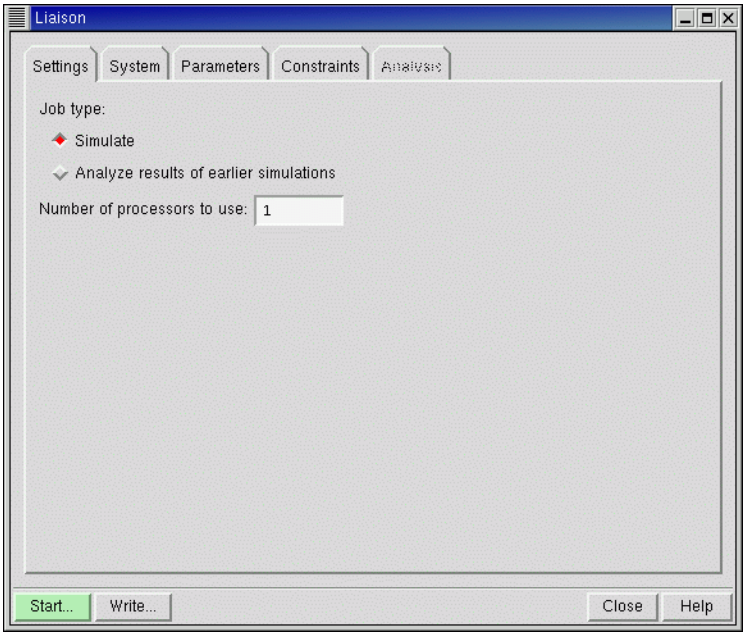


Figure A.1. The* Settings *folder of the* Liaison *panel.

- Analyze results of earlier simulations

  Select this job type to analyze the output of completed Liaison simulations by fitting calculated results to empirical binding energy values or by predicting the binding energy of new ligands. These jobs are set up using the options in the Analysis folder. This selection deactivates the System, Parameters, and Constraints folders and activates the Analysis folder.

Number of processors to use

Liaison simulation jobs can take advantage of multiple processors to perform distributed processing. Use this text box to specify the number of processors on which to run concurrent Liaison simulations. For example, if there are 10 ligand/receptor combinations (for a total of 20 jobs — 10 "free" and 10 "bound") and there are 8 processors, setting this number to 4 will launch 4 jobs when you click the Start button. On each processor, when one job completes, another job will start, until all simulations have been submitted.

Note: This option indicates how many processors on the same machine to run simultaneously; it is meaningless when jobs are submitted to a batch queue, where each of the ligands is independently queued.

This option is not available when the Job type is Analyze results of earlier simulations, as analysis jobs are much faster than simulation jobs.

A.4 The System Folder

This folder is available only when the Job type selected in the Settings folder is Simulate. Options in this folder set the type of simulation to be run and defines the system and the source of the ligand or ligands to be used. The key option menu is Simulation type, and the two choices offered are:

- Multiple ligands, single receptor
- Single ligand, single receptor

The choice of simulation type determines which options appear or are made available in the remainder of the System folder. Options for the multiple-ligands case are described in Section A.4.1; for the single-ligand case, in Section A.4.2.

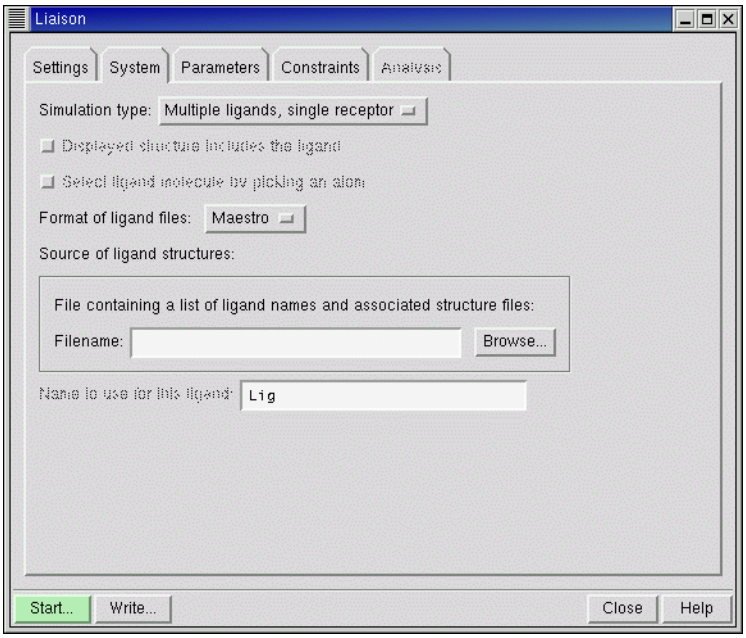


Figure A.2. The System ***folder of the*** Liaison ***panel.***

A.4.1 Multiple Ligands, Single Receptor

To run a simulation (Simulate task, Fit or Predict step) using two or more ligands, you must provide one receptor structure and the set of ligand structures.

The receptor structure is taken from the Workspace, which must contain this, and only this, structure. Receptor structures for use in Liaison should be prepared as described in Chapter 4.

To provide ligand structures, you must specify a text file listing the ligands to be used and the locations of the associated structure files. This text file can reside in any directory.

The following options are used to specify ligand structures:

Format of ligand files option menu

Liaison supports the three file formats listed below for reading structures from local or network-mounted disks. Liaison does *not* support Mol2 formatted files, nor will it accept structures with united atoms or explicit lone pairs, which are incompatible with the OPLS-AA force field. At a minimum, hydrogens must be added. See Chapter 5 for information about ligand preparation.

- Maestro: Maestro-written files (extensions `.mae`, `.out`, or `.dat`)
- MDL SD: SD-formatted files (extensions `.mol` for single structure files and `.sdf` for multiple structure files)
- PDB: Rutgers Center for Structural Biology Protein Data Bank files (extensions `.pdb` or `.ent`)

Source of ligand structures option

File containing a list of ligand names and associated structure files

The source of the ligand structures must be a user-created text file containing a list of ligand names and their associated structure files. The text file cannot have the same name as the Liaison job itself, because the job name is used to create the master Liaison run directory. If it does have the same name, Maestro displays an error message.

The name of the file containing the ligand names and structure-file locations can be entered directly into the Filename text box. Alternatively, the Browse button adjacent to the text box can be used to activate the Open File panel to aid in locating the file. The identity of the selected file is then displayed in the Filename text box.

The format of each row in the file is:

LigandName `[`*space*`]` *LigandLocation*

For example:

```
1bkm_3m_1   /home/user/structs/1bkm_3m_1.mae
1bkm_3m_2   /home/user/structs/1bkm_3m_2.mae
1bkm_3m_3   /home/user/structs/1bkm_3m_3.mae
```

The first column contains the (user-defined) ligand name, while the second gives the directory path and file name of the ligand structure file. Liaison uses each ligand name to create a correspondingly named ligand directory under the master Liaison run directory. Additional ligands can be added by entering new lines containing the ligand name and the location of the associated structure file. Only spaces, not tabs or commas, can separate the ligand name and file name.

A.4.2 Single Ligand, Single Receptor

A single-ligand simulation requires a receptor structure and a ligand structure. The receptor structure is taken from the Workspace. It may be alone in the Workspace or may be accompanied by the ligand to be simulated. Receptor and ligand-receptor complex structures for use in Liaison should be prepared as described in Chapter 4.

If the ligand structure is not in the Workspace, you can specify a file as the source instead. The selection options for this Simulation type are described below.

Displayed structure includes the ligand

Select this option if the ligand to be simulated is included with the receptor structure in the Workspace. The ligand and receptor structures may be loaded in any order, or they may be part of a single structure file. The receptor is defined as that part of the Workspace which is not the ligand molecule. Selecting this option also selects the option to Select ligand by picking an atom. The ligand molecule you pick will be the only ligand simulated.

If the structure in the Workspace does not include the ligand molecule, deselect this option. File specification options become available so that a file containing a single ligand structure can be identified. See Format of ligand file option menu below.

Select ligand molecule by picking an atom

When Displayed structure includes the ligand is selected, this option is selected by default to provide a means of distinguishing the ligand from the receptor. To define the ligand molecule, click on a ligand atom in the Workspace structure. The ligand is marked in dark green Ball & Stick markers. All atoms that are not part of the ligand molecule are treated as part of the receptor.

Note: If you do not identify the ligand in this manner, it is treated as part of the receptor. The "receptor" structure this creates has no room in the active site for any ligand, so no bound structures can be simulated and jobs cannot succeed.

Format of ligand file

Liaison supports the three file formats listed below for reading structures from local or network-mounted disks. Liaison does *not* support Mol2 formatted files, nor will it accept structures with united atoms or explicit lone pairs, which are incompatible with the OPLS-AA force field. At a minimum, hydrogens must be added. See Chapter 5 for information about ligand preparation.

- Maestro: Maestro-written files (extensions `.mae`, `.out`, or `.dat`)
- MDL SD: SD-formatted files (extensions `.mol` for single structure files and `.sdf` for multiple structure files)
- PDB: Protein Data Bank files (extensions `.pdb` or `.ent`)

Source of ligand structure options

File containing a single ligand structure

When the ligand is not in the Workspace (i.e., when Displayed structure includes the ligand is not selected), you can enter the location of the ligand structure file into the Filename text box, or click the Browse button adjacent to the text box and navigate to the file in the Open File file selector. The file name is then displayed in the Filename text box.

Name to use for this ligand

This text box specifies a user-assigned name for the ligand. The default is `Lig`. The name is used to create a *ligandname* directory under the Liaison *jobname* directory.

A.5 The Parameters Folder

This folder is available only when the Job type selected in the Settings folder is Simulate. Use this folder to specify the details of the simulation calculation.

Each Simulate task runs both a "free" (ligand only) simulation and a "bound" (ligand-receptor complex) simulation. Parameters such as sampling method, minimization algorithm, and force field are required to have the same settings for both simulations. The upper part of the folder contains these shared options. Other options can be set for each simulation independent of the

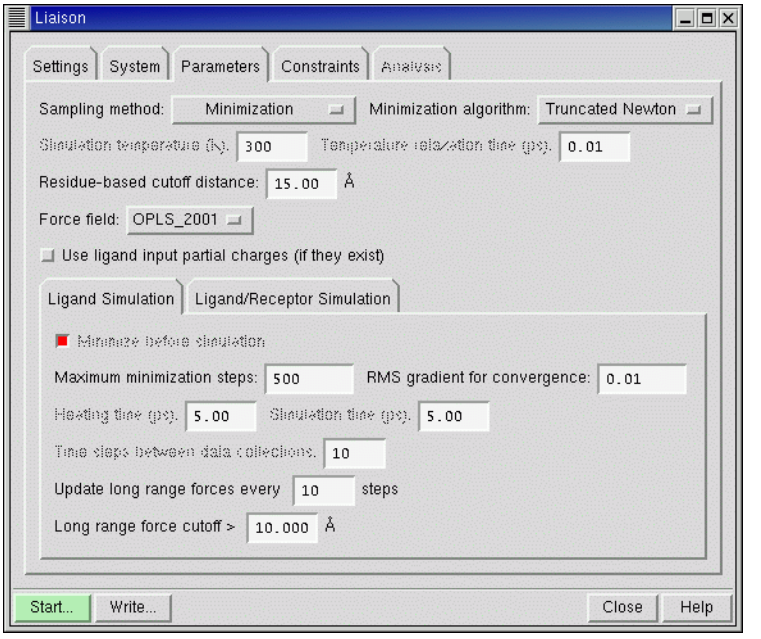


Figure A.3. The* Parameters *folder of the* Liaison *panel.

setting in the other. The lower portion of the folder is divided into two subfolders, Ligand Simulation and Ligand/Receptor Simulation, each with its own set of options, so that they can be set to different values.

A.5.1 Sampling Method

This section includes seven options, two of which are dimmed when Minimization is chosen as the Sampling method.

Sampling method

The supported sampling methods are:

- Minimization. This method performs an Impact energy minimization on the system. See Chapter 3 of the *Impact User Manual* for more information on this type of calculation.
- Hybrid Monte Carlo. This method employs the Hybrid Monte Carlo algorithm to sample the binding of the ligand to the receptor, or the conformation of the free ligand, depending on which simulation is being performed. See Chapter 5 of the *Impact User Manual* for details on the Hybrid Monte Carlo task and settings. When simulation (as opposed to minimization) is used for sampling, HMC is the recommended option.
- Molecular Dynamics. This method employs a Molecular Dynamics algorithm to sample the binding of the ligand to the receptor, or the conformation of the free ligand, depending on which simulation is being performed. See Chapter 4 of the *Impact User Manual* for details on the Dynamics task and settings.

By far the fastest method is energy minimization. Even though this method gives only a snapshot of the possible ligand-receptor configurations, studies to date have shown that it gives predicted binding affinities that are reasonably close to those obtained with HMC or MD. Minimization is thus an attractive choice when large numbers of ligands are to be studied. Those predicted to be most active might then be re-examined using a simulation protocol (HMC is recommended), if desired.

Minimization algorithm

This menu is available for all three sampling methods (there is an option to Minimize before simulation for the HMC and MD sampling methods.)

- Truncated Newton

  This is a very efficient method for producing optimized structures and is the current default selection. A short conjugate gradient pre-minimization stage is performed first to help improve the convergence of the Truncated Newton algorithm.

- Conjugate gradient

  This is a good general optimization method.

- Steepest descent

  This can be a good method for initiating a minimization on a starting geometries that contains large steric clashes. Convergence is very poor towards the end of minimization, where the Conjugate Gradient method should be used instead.

Simulation temperature

This option is available for Hybrid Monte Carlo and Molecular Dynamics sampling. It sets the simulation target temperature in Kelvin.

Temperature relaxation time

This option is available for Hybrid Monte Carlo and Molecular Dynamics sampling. It sets the time scale, in picoseconds, on which heat exchanges with the heat bath.

Residue-based cutoff distance

This text box sets the value for the cutoff distance. All pairwise interactions of an atom in residue *i* with an atom in residue *j* are included on the non-bonded pair list if any such pair of atoms is separated by this distance (in Å) or less. The default value is 15 Å.

Force field option menu

The force field options are OPLS_2005 (the default) and OPLS_2001.

Use ligand input partial charges (if they exist)

Select this option to use the input charges for the ligand in Liaison calculations (for both the free and the bound states).

Selecting this check box indicates that the partial charges in the input ligand Maestro files should be used instead of charges assigned by the force-field atom typer. If you have high-quality partial charges from, for example, ab initio electrostatic potential fitting, then this option can be useful.

A.5.2 Ligand Simulation and Ligand/Receptor Simulation

Depending on the sampling method chosen, some or all of the following options are available in the lower section of the folder. The two sub-tabs affect simulations on the free ligand (Ligand Simulation) and bound complex (Ligand/Receptor Simulation), respectively.

The selection options are:

Minimize before simulation

This toggle is active only for Hybrid Monte Carlo and Molecular Dynamics sampling. It places a minimization task (MINIMIZE) in the Liaison input file in front of the LRM simulation task. Its purpose is to ensure that the simulated structure does not have significant excess potential energy from bad internal contacts.

Maximum minimization steps

This text box sets the maximum number of minimization steps. This option is active only when Minimize before simulation is selected. The default is 1000 steps.

RMS grad for convergence

This text box sets the criterion on the rms gradient for convergence of the minimization (kcal/mol/Å). This option is active only when Minimize before simulation is selected. The default value is 0.01.

Heating time

In an HMC or MD simulation, this text box sets the time (ps) over which the system is heated before the LRM task is launched to obtain averages for the van der Waals, Coulombic, reaction field, and cavity terms. The default value for the heating time is 5 ps.

When the Liaison input file is written, the heating time is converted to the value of `mxcyc` (HMC) or `nstep` (MD) written in the HMC or Dynamics sections. The table below shows how the conversion is made.

Table A.1. Conversion formulae for heating time

Task	Conversion Formula
HMC	Heating Time = `mxcyc * nmdmc * delt *` 6
MD	Heating Time = `nstep * delt *` 6
Min	N/A (Liaison-Minimization jobs do not do heating at all)

The terms in the formulae are as follows:

`mxcyc` = # of HMC cycles
`nmdmc` = # of MD steps per MC cycle
`nstep` = # of MD steps
`delt` = time step (in ps). `delt` is 0.002 ps for HMC, and 0.001 ps for MD.

The default 0.002 ps time step for HMC and 0.001 ps time step used for Liaison dynamics (MD) jobs cannot be modified in Maestro, but can be edited by hand in the input files. However, this is not recommended. The factor of 6 comes from the fact that heating is broken up into six equal stages.

Example: HMC method, Heating time = 15 ps. Resultant input file (HMC section only):

```
-----------------------------------------------
HMC
input cntl mxcyc 83 nmdmc 5 delt 0.002 relax 0.01 nprnt 100 seed 101
-----------------------------------------------
```

The number of HMC steps (`mxcyc`) is five times smaller than the number of MD steps (`nstep`) because each composite HMC step includes five MD steps (set by `nmdmc 5` in the example above). Note also that the calculated number of steps (83) corresponds to one-sixth of the requested heating time. This is because Liaison heats the ligand-protein complex (but not the free ligand) in six equal temperature increments, each of which receives one-sixth of the total heating time. For example, for a target temperature of 300 K, the heating is done in 50 K increments of 0 – 50 K, 50 – 100 K, ..., 250 – 300 K.

Given that the Liaison panel has default heating time of 5.00 ps, time step of 0.002 ps, and number of MD steps per MC step of 5, then:

`mxcyc` = (5.000 ps / (0.002 ps * 5) = 500. 500 divided by 6 increments is 83 (rounded) steps per increment.

Simulation time

In an HMC or MD simulation, this text box sets the simulation time for the LRM task used to determine the averages for the van der Waals, Coulombic, reaction field, and cavity terms. The default value for the simulation time is 5 ps.

When energy minimization is used for sampling, no heating is done, but a short pro-forma HMC simulation (`mxcyc = 10`) is carried out at 10 K to obtain the needed “averages” for the Liaison interaction quantities.

When the Liaison input file is written, the simulation time is converted to the value of `mxcyc` (HMC) or `nstep` (MD) that is written to the LRM task. Table A.2 shows how this is done.

The terms are defined as for the heating time. The default 0.002 ps time step for HMC and 0.001 ps time step used for Liaison dynamics (MD) jobs are not modifiable inside Maestro, but can be edited by hand in the input files. However, this is not recommended.

When setting up Liaison jobs from Maestro, `delt` and `nmdmc` remain constant, while `mxcyc` and `nstep` are increased/decreased to accommodate the user-specified simulation time.

Table A.2. Conversion formulae for simulation time

Task	Conversion Formula
HMC	Simulation Time = `mxcyc * nmdmc * delt`
MD	Simulation Time = `nstep * delt`
Min	N/A

Example: Dynamics method, Simulation time = 5 ps. Resultant input file (LRM section only):

```
------------------------------------------------
sample DYNAMICS
input cntl nstep 5000** delt 0.001 relax 0.01 nprnt 100 seed 101
------------------------------------------------
```

** 5/0.001 = 5,000 steps

The five-fold reduction in the number of HMC steps (`mxcyc`) reflects the fact that each composite HMC step includes five MD steps.

Time steps between data collections

This text box sets the number of time steps between data collections of the ensemble averages during the Liaison sampling. Entering 10 in this box (the default value) produces a line like the following in the LRM task of the Liaison input file:

```
input cntl average every 10 file lia_free.ave
```

Update long range forces every *n* steps.

This option is available only for the Truncated Newton algorithm. The default is to update long range forces every 10 steps. Between updates, estimates of these forces are used. Smaller values of *n* (more frequent updates) can be used to improve convergence, but will make the optimization slower. The maximum recommended value is 20.

Long range force cutoff > *n* Angstroms.

This option is available only for the Truncated Newton algorithm, and specifies the distance beyond which forces will be treated as “long range”—that is, updated every *n* steps, as specified in the previous option, and estimated between updates.

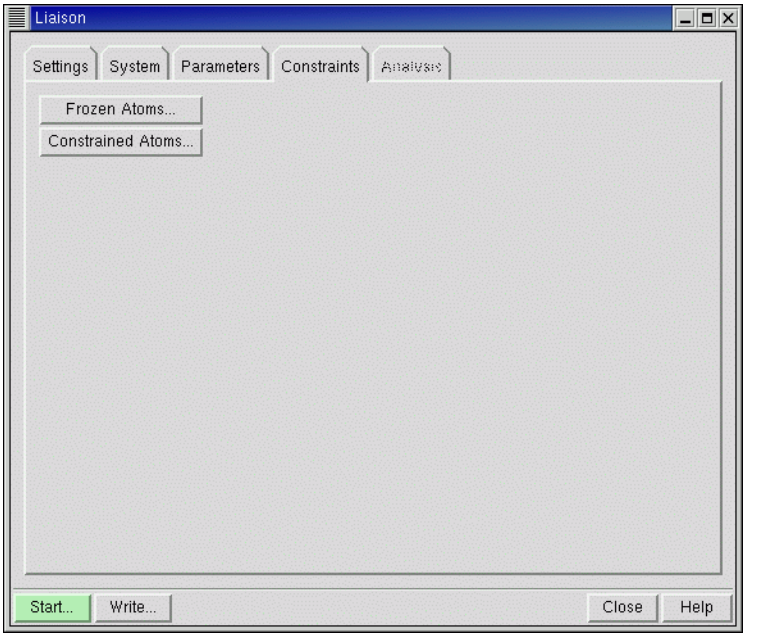


Figure A.4. The Constraints ***folder of the*** Liaison ***panel.***

A.6 The Constraints Folder

From this folder, available only for Simulate jobs, you can open panels that allow you to freeze or constrain selected atoms in the receptor. You can use these options to designate regions of freely moving atoms, constrained atoms, and fixed atoms: for example, allowing receptor atoms closest to the ligand to move freely while constraining a range of atoms at intermediate distance from the ligand and fixing (freezing) more distant atoms.

The selection options are described below.

Frozen Atoms

Liaison simulations can be performed with some atoms "frozen," so that they never move from their initial position during minimization or dynamics. Clicking this button opens the Frozen Atoms panel, which selects atoms of the receptor to be treated as frozen. The atoms or atom sets can be chosen by picking atoms, residues, or molecules from the Workspace, or by using the Atom Selection dialog box. See Chapter 5 of the *Maestro User Manual* or the Maestro online help for information on the Atom Selection dialog box. For more information about the Frozen Atoms panel, see the Maestro online help as well as Section 3.5.1 of the *Impact User Manual*.

Note: These frozen atom selections are keyed to the Workspace structure. Jobs will only include these constraints if they are run on the Workspace structure.

Constrained Atoms

Click this button to open the Constrained Atoms panel, which allows you to specify atoms or atom sets to be treated as "constrained", i.e., allowed to move but subject to harmonic penalty-function restraints. Along with the standard picking controls and the Atom Selection dialog box, the Constrained Atoms panel includes the Constraining force text box, which allows you to set the force constant for the force constraining the selected atoms. The same force constant is used for all atoms. The default is 25.00 kcal/(Å2 mol).

Note: These constrained atom selections are keyed to the Workspace structure. Jobs will only include these constraints if they are run on the Workspace structure.

A.7 The Analysis Folder

The Analysis folder contains the options for setting up jobs of type Analyze. This folder is available only when Analyze results of earlier simulations is selected in the Liaison Settings folder.

A Liaison analysis calculation uses the results of completed Liaison simulations to fit the binding energy model coefficients to the binding energies of known ligands or to predict the binding energies of new ligands. To use the results of earlier simulations, a Liaison Analysis job must be run from the same directory (Maestro working directory), and the same text string must be entered into the Job text box (to identify the master Liaison run directory). In addition, the supplied ligand names must be the same as those used in the simulation calculations.

To fit the results of simulation calculations using a training set of ligands with known binding energies (Analyze task, Fit step) you must specify a file that associates the training set ligands with their experimental binding energies.

To analyze simulations on test ligands to predict their binding energies (Analyze task, Predict step) you must specify a text file of ligand names or enter this information as a comma-separated list.

A.7.1 Analysis Settings Section

Use these options, in the upper part of the Analysis folder, to select the type of analysis to perform, the binding energy model to use, and optionally to constrain fitting or prediction parameters to values you enter. The options are:

Analysis type

- Fit. This option requires a text file containing at least three ligand names and the associated binding energies. The specification and format of this file are described under File name of ligand binding energies.
- Predict. This option uses values of the selected binding energy model's Fitting parameters together with data obtained from completed simulations, to predict the binding energy for a ligand or a series of ligands. The ligand names can be entered directly or can be taken from a user-created text file that lists ligand names with their structure files. The file format is described in Section A.7.2.

Binding energy model

Use these options to specify which binding energy equation to use for fitting and predictions. The two choices are:

- LIA equation

  This model has three parameters: van der Waals α, electrostatic β, and cavity γ, as shown in Section 1.2 on page 2.

- LiaisonScore

  This is a linear model which uses only slope (a) and intercept (b) fitting parameters to fit to experimental binding energies. See the LiaisonScore equation in Section 1.2 on page 2.

Fitting parameters: check box to constrain/Prediction parameters

These text boxes represent the coefficients for the selected binding energy model equation. For the LIA Equation, these are the coefficients of the Van der Waals, Electrostatic, and Cavity energy terms. For the LiaisonScore model, they are the Slope and Intercept coefficients. These values are expected to vary for each system studied; there are no "universal" default values.

When you start to set up a Fit Analyze job, the value in each text box is zero. Fit jobs normally derive values for all the parameters appropriate to the selected binding energy model. Optionally, however, you can set and constrain a value for one or two of the parameters (but not all). Select the check box next to the parameter you want to constrain. The text box becomes available. Specify the constraint value by typing in the text box. (Check boxes do not appear next to parameters not appropriate for the model.)

When you open the Analysis folder to set up a Predict job, the values derived (or constrained) in the Fit job appear in the text boxes under the heading Prediction parameters.

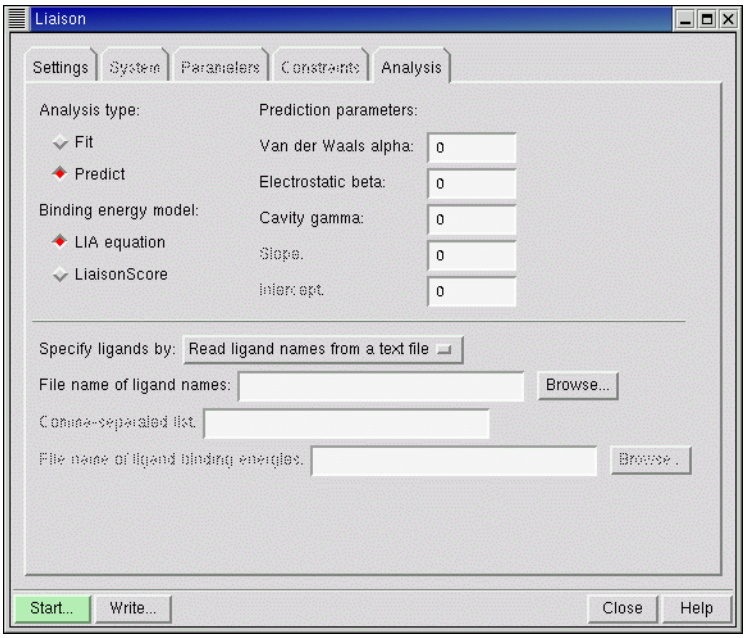


Figure A.5. The Analysis ***folder of the*** Liaison ***panel with analysis type*** Predict ***selected.***

A.7.2 Ligand Specification Section

Fit Step

For Fit step analysis jobs, this lower section of the Analysis folder is used to specify a file with a list of ligands and their experimental binding energies:

File name of ligand binding energies text box and Browse button

The file you specify must satisfy the following requirements:

- Only plain text files are permitted.
- Ligand binding energies should be provided in kcal/mol to match the output of the Liaison calculation. (Other choices, such as kJ/mol or pK_i, will give correct numerical results in the same units, but “kcal/mol” will still be printed.)
- The minimum number of ligands is 3, as there are 3 LIA equation parameters to fit. In actual practice, binding energies of at least 7 and ideally of 10 to 20 or more ligands should be provided for fitting.

For example, a file of three ligands with their experimental binding energies would look like this:

```
1bkm_3m_1   -10.5
1bkm_3m_2   -10.9
1bkm_3m_3   -11.9
```

where the first column contains the ligand names, the second column lists the experimental binding affinities, and each ligand name must be separated from its binding energy only by spaces.

The ligand names are also the names of the ligand directories `1bkm_3m_1/`, `1bkm_3m_2/`, and `1bkm_3m_3/`, which contain the simulation data needed by the Analyze job. These can be found in the master Liaison run directory, *jobname*`/`, where *jobname* is the name of the Predict step simulation job.

Predict Step

To specify ligands for Predict step analysis jobs, choose from the Specify ligands by option menu:

- Read ligand names from a text file
- Enter list of ligand names

Read ligand names from a text file

Choose this option if you want to specify a user-created text file that contains the names of the ligands whose binding energies will be predicted. The names can appear in the file as a comma-separated, space-separated, or carriage-return-separated list. For example:

```
1bkm_3m_1, 1bkm_3m_2, 1bkm_3m_3
```

These entries instruct Maestro to retrieve the requisite data from completed Liaison simulations from directories `1bkm_3m_1/`, `1bkm_3m_2/`, and `1bkm_3m_3/` under the Liaison *jobname* directory.

Enter list of ligand names

If only a small number of ligands are to be predicted, it may be more convenient to choose this option and use the Comma-separated list text box to enter a comma-separated list of ligand names. Spaces can also be used to separate names, but tabs cannot. For example:

```
1bkm_3m_1, 1bkm_3m_2, 1bkm_3m_3
```

A.8 Liaison Directories and Files

This section describes the directories and files that are created in the course of running Liaison.

A.8.1 Liaison Directory Structure

Figure A.6 shows a schematic overview of the Liaison directory structure, where "Maestro working directory" is the directory in which Maestro is running when you start a Liaison job or choose to Write job files.

Files in brackets [ ] are created only with Liaison Dynamics and HMC jobs.

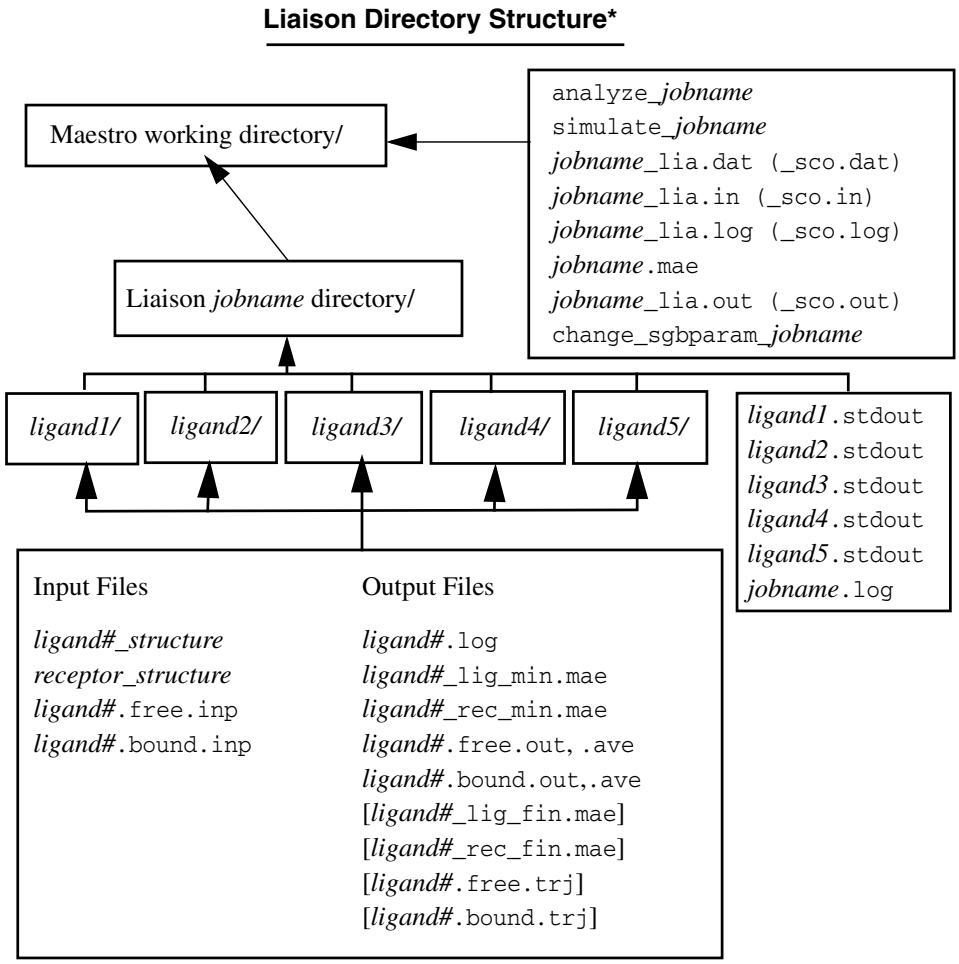


Figure A.6. The Liaison directory structure.

A.8.2 Directories Created

When a Liaison job is given a name (either when you start the job or when you write job files) a directory by that name is created in the Maestro working directory. All files related to the job are stored under this Liaison *jobname* directory.

If you are running a multiple-ligand simulation, for each ligand, a directory is created under the Liaison *jobname* directory. The names of the directories are defined by the ligand names that you specified in the text file.

A.8.3 Files Created

In the Maestro working directory, the following files are created:

`change_sgbparam_`*jobname*	Utility to modify the SGB solvation parameters for the input files in all the ligand subdirectories.
`simulate_`*jobname*	The main script Maestro uses to dispatch the Liaison simulation.
jobname`.mae`	The receptor (or receptor/ligand) structure file in Maestro format. This file is written by the Maestro interface.
`analyze_`*jobname*	Script to run a Liaison analysis (fitting or predicting) job. The job name *jobname* should be changed to distinguish the Predict job from the previous Fit job.
jobname`_lia.in`	Input file for a Liaison analysis job using the LIA binding energy model. If the LiaisonScore binding energy model is chosen, the file name is *jobname*`_sco.in`.
jobname`_lia.dat`	Data file for a Liaison LIA analysis job.
jobname`_lia.log`	Log file for a Liaison LIA analysis job.
jobname`_lia.out`	Output from a Liaison LIA analysis job.

In the Master Liaison Run Directory, the following files are created:

jobname`.log`	Log of ligands submitted.
ligand#`.stdout`	Of interest only if an error has occurred.

In the *ligand#* directory or directories, the following files are created:

jobname`.free.inp`	The Liaison input file for simulation of the free ligand.
jobname`.bound.inp`	The Liaison input file for simulation of the ligand-receptor complex.
jobname`_lig_min.mae`	Final minimization structure for the ligand from the bound simulation, when minimization is used as the sampling method or when the ligand is minimized prior to Hybrid Monte Carlo or Molecular Dynamics sampling.
jobname`_lig_fin.mae`	Final simulation structure for the ligand from the bound simulation when HMC or MD sampling is used.
jobname`_rec_min.mae`	Final minimization structure for the receptor from the bound simulation, when minimization is used as the sampling method or when the complex is minimized prior to HMC or MD sampling.
jobname`_rec_fin.mae`	Final simulation structure for the receptor from the bound simulation, when HMC or MD sampling is used.

The following links and files are also created:

- Link to the receptor structure file *jobname*`.mae` that Maestro wrote in the Maestro working directory. In a single ligand job, the ligand is also contained in the structure file if it was taken from the Workspace.
- Link to the location of the ligand file, which may or may not be the same as the receptor structure file.
- Other output files from the simulations, including energy output (`*.out`, `*.log`, `*.ave`) and trajectory files from sampling (`*.trj`).

Note: When both ligand and receptor structures come from the same file (*jobname*.mae), both ligand and receptor structures are included in the files *jobname*`_rec_min.mae` and *jobname*`_rec_fin.mae`; when the ligand structure comes from a different file from the receptor structure, these files contain only the receptor structure.

Copyright Notices

C and C++ Libraries for Parsing PDB Records

Index

120 West 45th Street
32nd Floor
New York, NY 10036

101 SW Main Street
Suite 1300
Portland, OR 97204

3655 Nobel Drive
Suite 430
San Diego, CA 92122

Dynamostraße 13
68165 Mannheim
Germany

QuatroHouse, Frimley Road
Camberley GU16 7ER
United Kingdom

SCHRÖDINGER.